FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Matching Omnidirectional Images in Indoor Environments

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Visual Computing

eingereicht von

Timo Kropp
Matrikelnummer 0627880

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: a.o.Univ.-Prof. Dipl.-Ing. Dr.techn. Robert Sablatnig
Mitwirkung: Dr. Roman Pflugfelder

Wien, 19.03.2013

(Unterschrift Verfasser)

(Unterschrift Betreuung)

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Abstract

Image matching relates to the problem of finding corresponding points in two images of the same scene. An unsolved problem in computer vision is to match images from sparsely textured scenes. Examples of such scenes are indoor environments with homogenous regions, e.g. walls, ceilings or floors. It is shown that wide field of view cameras, so called omnidirectional cameras, can improve the matching problem in sparsely textured scenes. Approaches for matching omnidirectional images were proposed since traditional matching methods like the Scale Invariant Feature Transform (SIFT) do not consider the non-linear distortion introduced by omnidirectional images, captured with catadioptric or fisheye cameras. In this thesis the keypoint detection and descriptor matching performance is evaluated and compared for SIFT, sRD-SIFT and SIFT on the Sphere (SIFT Sphere) in the context of indoor scenery. The performance of keypoint detection is estimated with the measure repeatability and recall vs. 1-precision is used to compare the descriptor matching performance of each approach. sRD-SIFT and SIFT Sphere are adaptations of SIFT, adding a model of non-linear camera distortion for images with a single viewpoint. The main contribution of this thesis is the estimation of the general performance of those approaches in relation to different image transformations, e.g. scaling, rotation, viewpoint changes and field of view changes, and in respect of sparsely textured and structured scenes. It is shown that SIFT loses invariance to rotation, scaling and viewpoint changes in matching omnidirectional images. SIFT Sphere is the sole approach examined which is invariant to such transformations. Further it is shown that the performance of SIFT, sRD-SIFT and SIFT Sphere in matching omnidirectional images with each other is superior to the matching of perspective to omnidirectional images, which is the case in hybrid camera networks. Overall it is concluded that all approaches examined have a complementary performance, which highly depends on the scene type and image transformation, but in general none of them can be identified to be more superior to the others.

Kurzfassung

Keypoint Matching bezeichnet das Problem vom Finden korrespondierender Punkte in zwei Bildern. Dies ist für Bilder von schwach texturierten Szene ein ungelöstes Problem in der Computer Vision. Derartige Szenen befinden sich z.B. innerhalb von Gebäuden, welche hauptsächlich aus homogenen Flächen wie z.B. Wänden, Decken oder Böden bestehen. Es wird gezeigt, dass Kameras mit einem besonders großen Blickwinkel, sogenannte omnidirektionale Kameras, das Matching Problem in diesen Umgebungen besser lösen können. Jüngst wurden neue Algorithmen vorgeschlagen um das Finden von korrespondierenden Punkten in omnidirektionalen Bildern zu verbessern. Herkömmliche Matching Verfahren, wie die Invariant Feature Transform (SIFT) für perspektivische Bilder ignorieren die nicht-lineare Verzerrung von omnidirektionalen Bildern, die mittels katadioptrischen oder Fischaugen Kameras erzeugt werden können. In dieser Arbeit wird die Leistung der Keypoint Erkennung und deren Zuordnung für SIFT, sRD-SIFT und SIFT on the Sphere (SIFT Sphere) im Kontext von Innenraumfotos untersucht und evaluiert. Die Vergleichsmaße der Auswertung sind Repeatability und Recall vs. 1-Precision. sRD-SIFT und SIFT Sphere sind Adaptierungen von SIFT, die ein geeignetes Model für die nichtlineare Bildverzerrung von Single Viewpoint Kameras verwenden. Das Ziel dieser Arbeit ist die Leistung von den genannten Verfahren im Zusammenhang mit unterschiedlichen Bildtransformationen wie Skalierungen, Rotationen, Änderungen des Standpunktes und Änderungen des Blickwinkels in schwach strukturierten und texturierten Bildszenen zu untersuchen. Dabei werden nicht nur die Matching Ergebnisse von omnidirektionalen Bildern untereinander, sondern auch die Ergebnisse von perspektivischen Bildern mit omnidirektionalen Bildern miteinander verglichen. Der zuletzt genannte Fall tritt bei sogenannten hybriden Kamera Netzwerken auf. Es wird gezeigt, dass nur SIFT Sphere bezüglich der untersuchten Bildtransformationen invariant ist. Die wichtigste Erkenntnis dieser Arbeit ist, dass die untersuchten Verfahren komplementäre Ergebnisse liefern, welche wiederum stark von der jeweiligen Szene und Bildtransformation abhängen. Es wird gezeigt, dass sich keines der untersuchten Verfahren in Hinblick auf die Matching Leistung von den Anderen deutlich absetzt.

Acknowledgements

I thank profoundly Roman Pflugfelder for providing support throughout every aspect of this thesis. He greatly assisted me by inspiring conversations. For steady and trustful support I want to thank deeply Robert Sablatnig. His critical advices kept me not to lose the right focus. I appreciate for having thousands of inspiring discussions with Georg Nebehay. I want to thank him, for giving me valuable support in understanding new concepts. Without his help many of the ideas presented in my thesis would not have been included. I am fortunate and grateful to my colleague and good friend Richard Egan who did not only provide me with a sharp and critical eye, but who invested time correcting earlier drafts of my thesis and demonstrated his endless patience. I am also grateful to Branislav Micusik and Peter Gemeiner for providing their fundamental knowledge and for sharing their experiences within this research field. Thanks to Andreas Zoufal for giving me best wishes during the final stage and for being a nice roommate.

I am deeply grateful for having my parents and my sister, Jana, who never stop believing in me and my abilities. They give me motivation, encouragement and inspiration throughout. Finally, I want to thank Natascha Husar for her patient, her sympathetic ear and her critical judgment for supporting me in improving my scientific communication skills.

Contents

CHAPTER 1

Introduction

The human brain needs to constantly match the information perceived by the retina of each eye [16] in daily life. This is processed seemingly without any consciously effort, since it is done automatically. But in fact, even for the human brain this is a complex task [35]. Inspired by the human visual system, matching of digital images is tried to be solved with the detection of corresponding locations. Since the majority of digital images are captured with perspective cameras, algorithms for image matching in computer vision are adapted for this class of camera. Similar to the human eye [16], there are cameras with a field of view of approximately 180 degrees. In contrast to perspective cameras, they do not provide a constant spatial resolution and thus traditional matching algorithms are not capable to handle them properly. New approaches for image matching have been developed [8], [28] taking the specific geometry of wide field of view cameras into account.

1.1 Problem Definition

Image matching in computer vision relates to the problem of automatically identifying points or regions in two images of the same scene which refer to the same 3D location of that scene, as shown in Figure 1.1. Those points are called *correspondences*, as they correspond to the same scene position. Finding correspondences in two or multiple images is a fundamental concept in computer vision, as it has a variety of applications, e.g. object recognition [34], texture recognition [24], content-based image retrieval [25], object tracking [7], wide baseline matching [22], [48], 3D reconstruction [45], [13], structure from motion [30], outdoor localization [17], robot localization [39], video data mining [41] or camera calibration [42]. Tuytelaars et al. provide an extensive survey on interest point detection [46]. The matching problem is one of the fundamental problems in computer vision [31]. The core demand on point matching is that a point of interest in one image should be re-detectable and found in another image under image transformations such as viewpoint changes, scale changes or rotation changes. The optimal case is, if this point is matched as long as it is captured in the second image, regardless from which position, orientation or photometric conditions the image is taken [40]. Therefore the requirements of the region around the so called *interest point* or *keypoint* are to be *distinctive* and simultaneously *robust* to the aforementioned transformations [31].

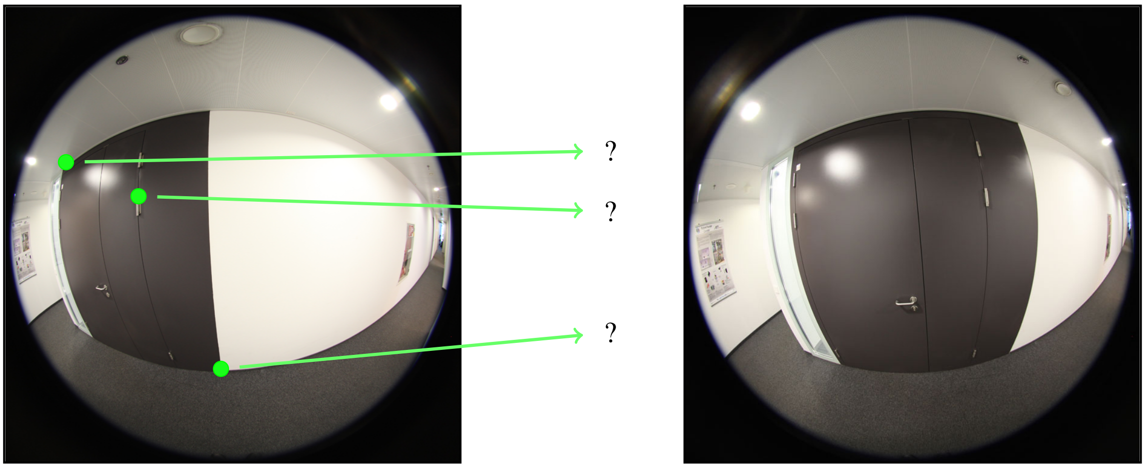


Figure 1.1: Identifying corresponding points in two images of the same scene.

In particular, environments with sparsely textured surfaces are still an open research field of image matching [10], since robust keypoints are relying on the image texture and structure. Indoor is a characteristic environment, where the major elements are homogenous regions from walls, ceilings, floors. For example, video surveillance [36] or robot localization [43] are typically faced with those scenarios.

This work proposes and shows that indoor matching can be improved by using large field of view cameras, so called *omnidirectional* cameras. A detailed characterization is given in Chapter 2. The increased viewing angle in contrast to narrow field of view cameras enables the detection of more robust interest points than in the other case, since more spatial structure is observed.

In the case of matching perspective images, the *Scale Invariant Feature Transform (SIFT)* [29] is a well examined and very successful approach. This approach is not only used by several authors in different applications [52], [38], [3] or [26], but is also evaluated to give the most reliable matching results compared to other state-of-the-art methods for matching, as shown in [31]. Unfortunately, as shown in [28] the matching performance of SIFT is not constant over images with different radial distortion. This is due to the fact that SIFT assumes images with constant spatial resolution. However, it is an open question how strong the matching performance is in general influenced by using omnidirectional images, since the distortion decreases in direction from the image borders to the image center.

Because of the lack of an adapted omnidirectional model in the case of SIFT [28], newly developed methods try to fill this gap by using a spherical image model or a specific non-linear distortion model for example [8], [28]. One of those is *SIFT on the Sphere (SIFT Sphere)* proposed by Cruz-Mota et al. [8]. Another approach to solve this problem is published by Lourenco et al. with the name *sRD-SIFT* [28]. The authors evaluated their methods using only a small set of different scenes, i.e. up to 3 image sequences, and not with respect to all geometric image transformations for which the original proposed SIFT is invariant. These are scale, rotation and viewpoint changes of < 50 degrees. For evaluation, they also mainly used artificial image data. Therefore it is unknown how well these methods perform on real world image matching, especially on images with sparse structure and texture, and in comparison to SIFT, when applied on omnidirectional images.

1.2 Related Work

sRD-SIFT assumes a radial distortion model for the description of the image distortion. Its most important contribution is that all operations of the algorithm are taking place on the original image data and no resampling of the image is needed. Lourenco et al. propose that interpolation is an important issue in performing SIFT on distorted images [28]. In contrast to sRD-SIFT, SIFT Sphere warps the original image data onto a spherical surface. Cruz-Mota et al. assume that a precise image model is more important than resampling the original image data [8]. Since there is a one-to-one mapping between omnidirectional images and the surface of the unit sphere, the sphere can be used as a model for all types of omnidirectional images from different kinds of cameras such as catadioptric or fisheye cameras.

To find scale invariant interest points in an image, SIFT first computes the *scale space* of an image [29]. Koenderink gives in [23] a fundamental theoretical background of scale space in computer vision. Based on that, Lindeberg shows how to estimate the scale space of an image, and what applications it can have [27]. The scale space has the objective to represent an image in different levels of detail. Before a SIFT-based keypoint detector and descriptor on the spherical image domain is proposed by Cruz-Mota et al., the key aspect, i.e. estimation of the scale space for spherical images, is published by several authors. Bülow proposes a spherical Gaussian kernel to filter the image in the spherical image domain, namely the Green's function of the spherical diffusion equation, see [6] for details. Gaussian filtering is a key processing step of estimating the scale space of an image. Daniilidis et al. go a step further than Bülow and not only estimate the spherical scale space, but also propose a method to estimate the optical flow of omnidirectional images without resampling the original image data [9]. A different approach to estimate scale space and interest points is given by Briggs et al. [4], this based on the assumption of one dimensional omnidirectional images. In contrast to Gaussian filtering, Puig et al. propose a method based on the Laplace-Beltrami operator to estimate the spherical scale space of an image [37]. This approach does not require resampling the image data.

To solve the sparse structure matching problem for perspective images, Dickscheid proposes a generic statistical model for the seamless integration of different specific existing matching approaches [10]. This relies on the assumption that for each specific problem instance at least one concrete matching method is already available. Scaramuzza et al. propose an approach to match omnidirectional images in the context of robot motion estimation [39]. This work is based on the extraction of vertical lines in catadioptric images and their description for matching. Goedeme et al. invent a matching framework for omnidirectional vision to automatically build an environment map for real-time localization in autonomous mobile robot navigation [15]. Not based on interest point extraction but on interest region detection, Mauthern et al. propose a method to match omnidirectional images while extracting virtual perspective camera planes

[44]. Arican et al. propose a formal derivation based on the heat diffusion equation to describe omnidirectional images with Riemannian geometry and to compute scale-invariant features [1]. An extensive survey on general omnidirectional sensing is given by Yagi in [50].

Not only the matching methods for perspective and omnidirectional images are established, but their evaluation is also a fundamental research topic. In case of perspective matching an extensive evaluation of state-of-the-art affine region detectors is done by Mikolajczyk et al. [32]. They estimate not only the performance of different detectors, but also the matching performance in relation to different region descriptors in [31].

The two specialized methods SIFT Sphere and sRD-SIFT are selected in this thesis for evaluation because they give two contrary state-of-the-art methods for matching omnidirectional images. SIFT Sphere formulates the entire process of keypoint detection and keypoint description in the spherical domain, i.e. the original image data is transformed onto the unit sphere. Hereby, a proper camera model of the given omnidirectional image is known a priori. Scale space estimation is done using the *Spherical Fourier Transform*, see [8] for details. SIFT Sphere includes two types of descriptors for detected keypoints. One is called *Local Spherical Descriptor (LSD)* [8], it is proposed to match omnidirectional images with each other and the other is called *Local Planar Descriptor (LPD)* [8] which is specialized to match perspective images with omnidirectional images. In this work, both types are compared with each other.

sRD-SIFT adapts the *division distortion model* proposed by Fitzgibbon [12]. A mathematical approach is used that enables the scale space to be estimated directly from the original image data. Since the division model is proposed for perspective images with radial distortion, it cannot describe the entire image plane of omnidirectional images which have a field of view more than 180 degrees. Nevertheless, because of its partial correctness, Lourenco et al. proposed sRD-SIFT for the use with fisheye images as well, which typically have this large field of view.

For this work, the general idea from Mikolajczyk et al. [32] is taken, to develop a robust and accurate evaluation framework for omnidirectional image matching. The basic idea is to take images of planar scenes for which a linear transformation between two images can be estimated. This transformation, i.e. the ground truth, enables predicting where a keypoint from one image should appear in a second image. Then it is possible to determine whether a keypoint is re-detected or not. Not only the re-detection can be estimated from the ground truth, but also if the matching, i.e. the association of keypoints from two images, is correct.

1.3 Scope of Work

In this thesis SIFT, SIFT Sphere and sRD-SIFT are evaluated and compared in terms of keypoint detector performance and descriptor matching performance. The evaluation is done with images of indoor environments where only sparsely textured and structured surfaces are available. Each scene type is assessed for all Euclidean image transformations against a planar object, i.e. scale changes, field of view changes, viewpoint changes and rotation changes. Not only omnidirectional images are matched with each other, but also perspective images with omnidirectional ones. Two evaluation criteria are used to estimate the performance of the keypoint detectors and the matching performance. In the first case this is the so called *repeatability* which is the relative number of re-detected keypoints. For the latter case, the *recall vs. 1-precision* measure is used.

It is based on the fact, that a matching algorithm can associate a matchable or a not matchable keypoint, correctly or incorrectly with another keypoint.

1.4 Contribution

The main contribution of this thesis is the estimation of the general performance of SIFT, sRD-SIFT and SIFT Sphere over all Euclidean image transformations and over sparsely textured images.

Since SIFT is originally proposed to be invariant to image rotations, scale changes and viewpoint changes up to 50 degrees in the case of perspective images, it is shown how the performance is influenced under those transformations by omnidirectional image matching.

From a theoretical point of view, omnidirectional images include an additional invariance to every camera rotation, if the sphere is used as the underlying image model [17]. This transformation type is included in the evaluation and it is shown how the different methods can keep invariance.

Since SIFT, sRD-SIFT and SIFT Sphere are based on different image models, a detailed performance analysis regarding the aforementioned transformations and scene types is given for each. It is shown that all methods act complementary and none are, in general, superior in omnidirectional matching. Furthermore, the pros and cons of each method are investigated and it is shown for which scene type and image transformation each approach gives the best results.

Additionally more specific issues in terms of omnidirectional image matching are analyzed. Wide field of view images afford the opportunity to match images with a larger field of view change than perspective images. It is estimated how the matching with SIFT, sRD-SIFT and SIFT Sphere is influenced by this transformation type. Since the resolution density is larger in the central region than at the borders of omnidirectional images, it is analyzed to check if this fact has an impact on keypoint detection and descriptor matching, especially for SIFT.

By definition, it is true that larger field of view cameras can observe more scene structure. In the case of matching images with sparse structure, the use of omnidirectional images should improve the results. The actual effect is estimated with the comparison of matching omnidirectional images with each other and perspective images with omnidirectional images.

Since the overall matching performance is examined, the specific performance of SIFT is estimated too. Here it is analyzed which specific requirements SIFT has on the scene, since its matching performance of omnidirectional images is unknown thus far.

Last but not least, it is shown that wide baseline matching even with the use of omnidirectional cameras cannot be improved.

1.5 Organization

The thesis is organized as follows. First, a fundamental theoretical understanding of omnidirectional vision is given in Chapter 2. Chapter 3 leads to image matching in general and matching of omnidirectional images in particular. SIFT, sRD-SIFT and SIFT Sphere are introduced and their keypoint detector and descriptors are discussed separately. Chapter 4 states the experimental setup of the evaluation. In this chapter not only the image sets but also the performance measures are introduced. Finally, Chapter 5 contains the actual results and a discussion of the entire experimental evaluation of SIFT, sRD-SIFT and SIFT Sphere in omnidirectional image matching. Chapter 6 concludes the evaluation and outlooks future work.

CHAPTER 2

Omnidirectional Imaging

In this chapter the theory of omnidirectional imaging is described. It starts with the discussion of what omnidirectional vision actually is, and how perspective images are related to omnidirectional vision. The chapter is concluded with the unified model for all types of central omnidirectional cameras. Not only the basic concepts are discussed, but the types of actual cameras for omnidirectional vision are also presented. All concepts of omnidirectional vision, which are used in the following evaluation, are described in the following sections.

This chapter is organized as follows. In Section 2.1 the shared theory of image formations in perspective and omnidirectional vision is given. A definition of different image domains follows in Section 2.2. The image domains are the interface between the camera and the matching algorithm. They describe the image format of different cameras and simultaneously they represent the image structure assumed by an algorithm. In Section 2.3 different camera types for capturing omnidirectional images are discussed. The last Section 2.4 describes how all different types of omnidirectional cameras can be merged into one single formal projection model.The geometrical concepts of the unified model of central catadioptric and fisheye cameras are given.

2.1 Image Formation

An image of the physical world can be captured through projecting rays of light with a single center of projections onto an arbitrary surface. Thus, theoretically it is possible to see the entire surrounding environment from one single viewpoint, as shown in Figure 2.1. Such a field of view in all directions of a sphere is called *omnidirectional* [33], since rays of light are captured from all directions regarding a certain point of view. With a single center of projection the single viewpoint constraint is fulfilled and thus from any omnidirectional image, pure perspective images can be obtained by projecting the rays of light onto a plane with an arbitrary distance to the center of projection (*focal length*) [33] as shown in Figure 2.1. The distance and the size of that plane defines the actual angle of the field of view. It follows directly that there is no perspective projection with a field of view equal to 180 degrees or larger, since the rays of light would run

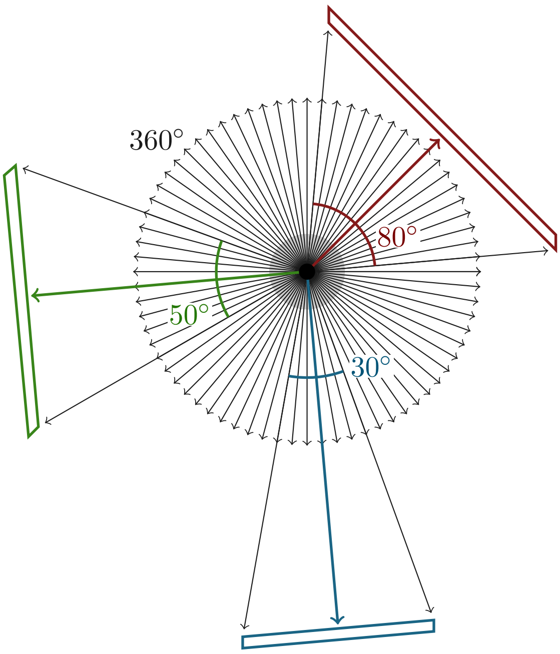
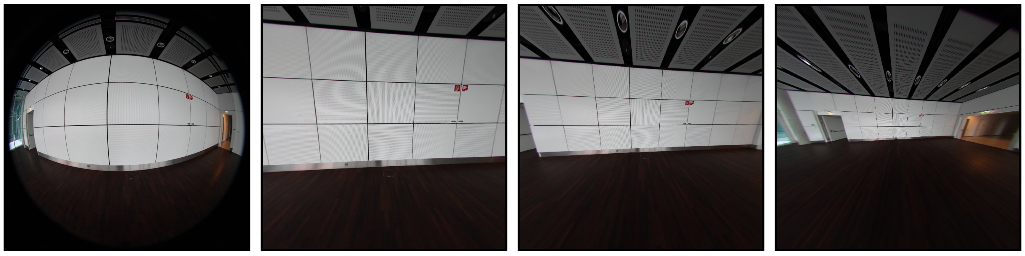


Figure 2.1: Fully omnidirectional vision from a single viewpoint (center of projection). Colored parts represent perspective projections with different fields of view.

parallel to that plane. The perspective projected images preserve linear geometry. Thus, straight lines in space are still straight on the image plane, for details see [19]. Nayar propose that there are two reasons for obtaining perspective images. First they are consistent with the vision system of the human eye and second the large body of work in computer vision assumes linear perspective projected images [33].



180° 120° 140° 160°

Figure 2.2: Omnidirectional image with 180° field of view (left). Undistorted perspective projections with different viewing angles. Note the increasing perspective distortion for larger fields of view.

As long as the single viewpoint constraint is fulfilled, perspective images can be obtained from omnidirectional images, and because of the large field of view it is still reasonable to capture omnidirectional images. For simplicity in this thesis, any image with a field of view larger or equal to 180 degrees is called *omnidirectional*, regardless from which camera it is obtained. Certainly it cannot be a perspective camera.

Images from perspective projection suffer from *perspective distortion*. In Figure 2.2 perspective projections from an omnidirectional image with different fields of view are depicted. It can be clearly observed that increasing the field of view results in stronger perspective distortion. For example, the panels of the ceiling are stretched more in the viewing direction with 160 degrees field of view as with 120 degrees. In reality their shapes are approximately quadratic. The omnidirectional image does not contain such strong perspective distortion even for a field of view of 180 degrees. This is due to the fact that rays of light are not perspectively projected onto a planar surface, but correspond to a projection onto a spherical surface. Finally, non-linear distortion enables to project this image onto a plane. This latter distortion type is called *radial distortion* or *lens distortion* of an image and must be distinguished to perspective distortion.

2.2 Image Domains

Depending on the type of camera, different image domains can be figured out. These are the image modalities which the matching algorithms assume as the input. It is distinguished between omnidirectional and perspective cameras. In the case of omnidirectional images, the data can be represented in spherical coordinates or similar to perspective images, in planar image coordinates. In Figure 2.3 the different image domains are shown as an illustration and with a corresponding sample image.

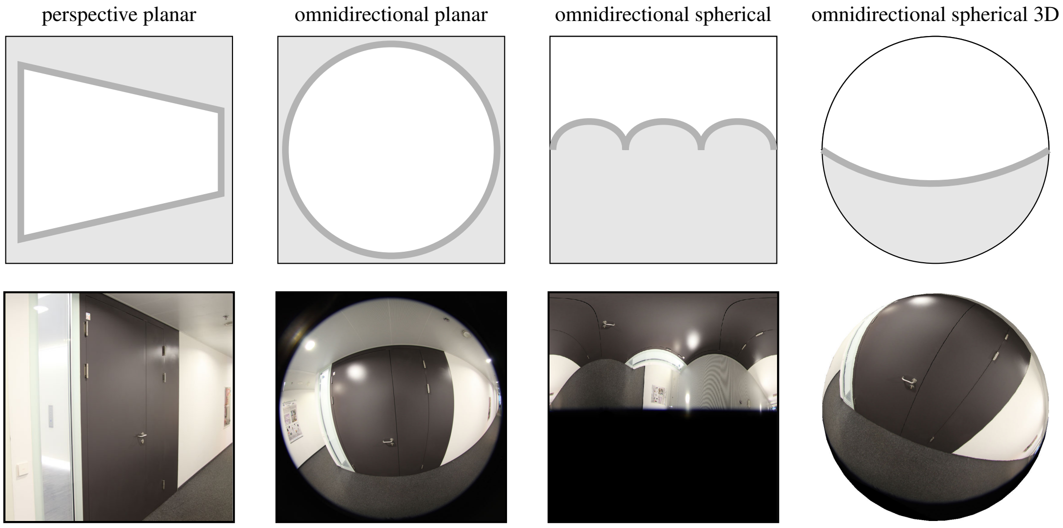


Figure 2.3: Image domains: illustrations and samples.

The term planar refers to the property of planar pixel representation. On the contrary, spherical relates to the case that the pixel positions are equal to a spherical angle and therefore can be represented on a sphere. In Figure 2.3 the pixels in spherical coordinates are shown with the label *omnidirectional spherical* and the actual representation on the sphere is shown with the label *omnidirectional spherical 3D*.

For all image domains the major difference is the actual pixel representation. It is assumed, that the sampling is uniform in each domain in relation to the actual spatial representation, but the underlying projected scene is represented arbitrarily. Planar perspective images without any image distortion have a constant resolution in spatial directions of the perspective image plane. This property is not valid for planar omnidirectional images, since non-linear distortion causes displacements of the pixel positions. For omnidirectional spherical images the resolution is equiangular, because each pixel position represents a certain angular coordinate of the sphere.

Matching algorithms, i.e. SIFT, sRD-SIFT or SIFT Sphere, are proposed on different image domains, e.g. SIFT assumes a planar perspective image [29], sRD-SIFT assumes a radial distorted image [28] and SIFT Sphere assumes an image in spherical coordinates [8]. In principle, that means the methods do not work in different image domains, because the actual pixel sampling is arbitrary. Nevertheless, images from calibrated cameras can be transformed at least partially to any other image domain. Warping the image means interpolation and introduces sampling errors, i.e. removing or adding high signal frequencies [28]. Therefore, Daniilidis et al. propose that in omnidirectional vision the original sampled values should always be used for any signal processing [9].

2.3 Cameras

After showing how perspective and omnidirectional images are obtained and what their underlying image domains are, now the actual camera types for perspective and omnidirectional vision are discussed.

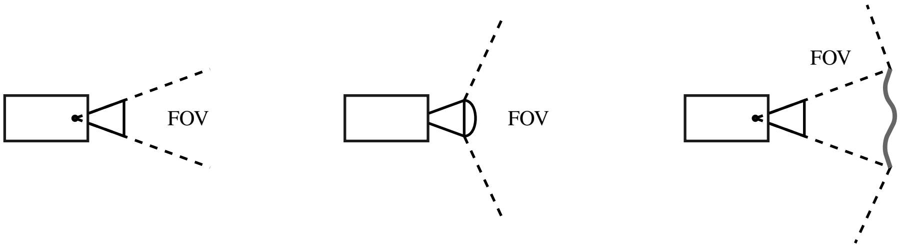


Figure 2.4: Illustrations of a perspective camera with a small field of view (left), a fisheye camera with large field of view (middle) and a catadioptric camera with large field of view (right). FOV = Field Of View. Image is taken from [33].

A perspective camera, as shown in Figure 2.4 (left) has a limited field of view in comparison to omnidirectional cameras, but it is consistent with the vision system of the human [33] and therefore produces familiar images. In contrast, omnidirectional images can be obtained by multiple different camera types. Yagi gives a survey of various ways to capture omnidirectional images [50]. Two of them are either the use of one rotating perspective camera or to bundle multiple perspective cameras in a way that the optical centers are coinciding. In practice this is hard to achieve [50]. Since these two solutions only obtain an omnidirectional image composed of multiple perspective images, they are not elaborated more in this work.

A wide angle view can be acquired by using a perspective camera with an attached, so called, fisheye lens, as shown in Figure 2.4 (middle). The field of view can cover the entire hemisphere, i.e. 180 degrees [50]. This special type of lens is designed in a way to capture rays of light which incide perpendicular to the optical axis. Another solution to enhance the field of view of a perspective camera is to place a reflective surface in front of it, as shown in Figure 2.4. Here, the incident angle of light can be increased by an appropriate mirror. Most importantly for such a construction is, that all incoming light rays are reflected so that they intersect in one single center of projection [33]. Otherwise the image obtained does not satisfy the single viewpoint constraint and no perspective images can be reproduced from that image. The optics of refracting elements (lenses) is called *dioptrics* and the optics of reflecting surfaces (mirrors) is called *catoptrics*. The combination of both elements is called *catadioptrics* [20], and therefore a camera with an attached reflecting surface is called a *catadioptric camera*. Baker et al. have proven that a catadioptric camera has a single effective viewpoint if and only if the mirror's cross-section is a conic section [2]. Practically, that means the shape of the mirror needs to be a rotated conic section to achieve a catadioptric camera with a single center of projection.

2.4 Unified Model

Every camera type, such as fisheye or catadioptric with an elliptical mirror, a hyperbolic mirror or a parabolic mirror, for example, produces images with different radial distortion. Therefore, those images are not practical in applying algorithms which are influenced by such distortions. If one single common projection model can be found, then algorithms can be adapted to this underlying shared model and would work with every omnidirectional image that is consistent with the single viewpoint constraint. This postulation is accomplished for all catadioptric images with a single viewpoint by Geyer et al. [14]. They prove that all cases of mirroring surfaces, e.g. parabolic, hyperbolic, elliptic or planar, with an appropriate orthographic or perspective camera can be modeled with a projection from world points to the unit sphere. And secondly, a projection from the sphere to the image plane. Here, the projection center is on a diameter of the sphere, to which the image plane is perpendicular.

As shown in Figure 2.5, the unifying theory from Geyer et al. contains two separated projections. Let $P = (x, y, z, w) \in \mathbb{P}^3$ be a point in 3D projective space, i.e. represented with homogenous coordinates, then P is projected onto the surface of a unit sphere S^2 centered at the origin $(0, 0, 0)$ by

$$F : \mathbb{P}^3 \rightarrow S^2 / \sim \tag{2.1}$$

$$F(x, y, z, w) = (\pm\frac{x}{r}, \pm\frac{y}{r}, \pm\frac{z}{r}) \tag{2.2}$$

with $r = \sqrt{x^2 + y^2 + z^2}$. The equivalence relation of antipodal points on the sphere is represented by $\sim$. To finally map the points from the spherical surface to the image plane $z = -m$, they are projected from $(0, 0, l)$. The mapping is done by

$$G : \mathbb{P}^3 \rightarrow \mathbb{R}^2_{\sim} \tag{2.3}$$

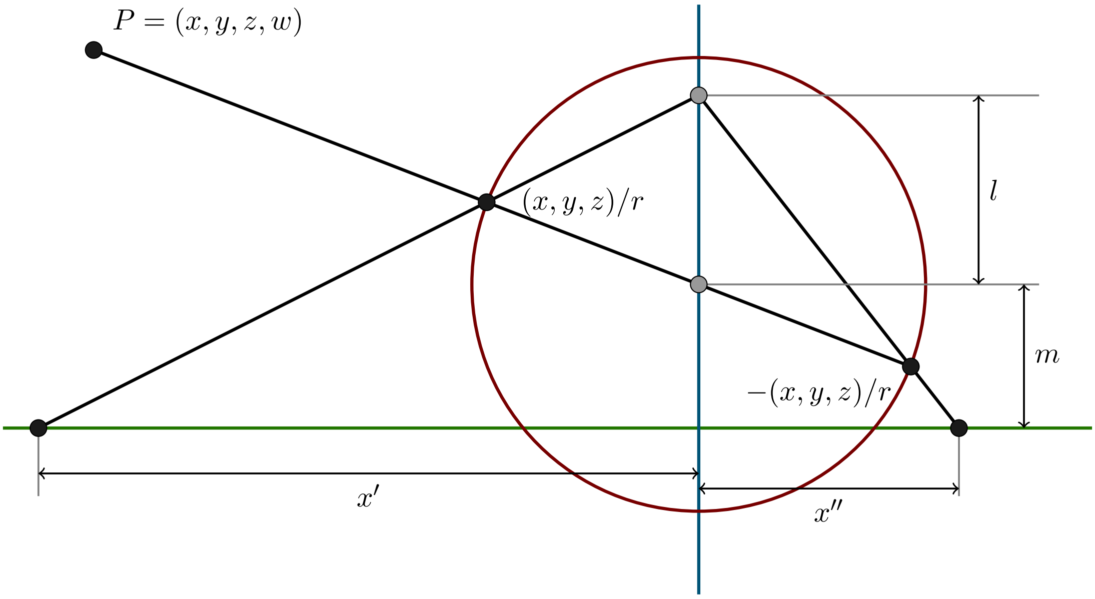


Figure 2.5: Unifying projection model. An arbitrary point P is projected onto the sphere (red) to the two antipodal points $(\pm x, \pm y, \pm z)/r$. Then both antipodal points are projected with the point $(0, 0, l)$ to the image plane (green). Image is taken with slight modifications from [14].

$$G(x, y, z, w) = (\pm\frac{x(l+m)}{lr \mp z}, \pm\frac{y(l+m)}{lr \mp z}, -m). \tag{2.4}$$

In Figure 2.5 the image plane is shown as a green line with $z = -m$. Thus, the principle axis (blue) is perpendicular to the image plane, passing through the center of the sphere and is coincident with the z-axis of the coordinate system. Likewise, the center of the perspective projection lies on that axis too, with $z = l$.

As shown in [14], changing m, i.e. moving the image plane in direction of z, results in scaling of the final image. Therefore $m = 1$ can be assumed. Similarly it is shown that special cases of $l = 1$ and $m = 0$ relate to the stereographic projection (the center of projection is at the North Pole) and $l = 0$ with $m = 1$ corresponds to a perspective projection [14].

Based on this projection model, Geyer et al. proved for all catadioptric projections with a single effective viewpoint that they are equal to the projection of a sphere followed by a projection to an image plane, as shown above [14]. This means they have proven the equivalence for projections with a parabolic, hyperbolic, elliptical and planar mirror. Each type relates to a different value of l in the range $[0, 1]$.

An extension to the unified model by Geyer et al. proposed is presented by Ying et al. [51]. Their extension allows describing not only central catadioptric images but images captured by fisheye cameras too. Hence a fisheye image can be transformed into a central catadioptric image and vice versa. In [51], it is shown that allowing $l > 1$, or $l = \infty$ in the unified model of Geyer et al. leads to the equivalence of the projection of fisheye cameras.

Finally, with the unified model of central catadioptric cameras and fisheye cameras an underlying shared geometrical surface is found, which is the unit sphere. All the various projection types can be described by the projection of points on a spherical surface to the actual image. It follows that if an image processing algorithm is proposed for spherical image data, then central catadioptric images or fisheye images can be processed. One way is to map the original image data onto the sphere before applying the algorithm. If resampling of the original image data needs to be avoided, then an adaption of the algorithm for the actual camera type is essential.

2.5 Summary

In this chapter the theoretical basis of omnidirectional vision was introduced. It was shown how omnidirectional and perspective images are captured and what their underlying image domains are. After introducing different camera types for omnidirectional imaging, the unified model from Geyer et al. was explained. It combines different types of cameras into a unified geometrical formulation.

CHAPTER 3

Matching of Omnidirectional Images

This chapter describes the matching approaches evaluated in this work. All of the approaches are SIFT based, which means they are adapted versions of the original SIFT proposed from Lowe [29]. These adaptations take the omnidirectional geometry into account. Specifically, SIFT Sphere and sRD-SIFT are considered. The former is a SIFT variation for image matching of spherical images and the latter approach uses the division distortion model [12] for keypoint and descriptor calculation.

First in Section 3.1, the general matching approach of SIFT is shown. All basic parts of the processing pipeline are discussed and decomposed into elementary concepts, which are further exchanged in the cases of SIFT Sphere and sRD-SIFT. Since keypoint matching relies on the part of keypoint detection and descriptor estimation, those stages are presented for each matching approach in Section 3.2 and Section 3.3 separately. Finally, the actual descriptor matching is presented in Section 3.4, which is the same for each matching method.

3.1 Matching Approach

Finding corresponding points in two images is in the case of SIFT, a three step approach. As shown in Figure 3.1 it is split into *keypoint detection* of an image, *descriptor estimation* of each keypoint and finally the *matching* of each keypoint with its corresponding descriptor. Keypoints are point locations in an image which relate to a distinctive local image structure, e.g. corners, gradients. This is necessary so that they can be uniquely identified. In contrast, descriptors are based on image regions, i.e. sets of multiple pixels, and abstract their specific content in a numerical representation. They are a *description* of a certain region. The detection of keypoints makes it possible to find potential corresponding image locations and the search of similar descriptors enables the assignment of those locations.

Let $P^1 = \{\mathbf{p}_1^1, \mathbf{p}_2^1, ..., \mathbf{p}_M^1\}$ and $P^2 = \{\mathbf{p}_2^1, \mathbf{p}_2^1, ..., \mathbf{p}_N^1\}$ with $\mathbf{p}_i^k = (x_i^k, y_i^k) \in \mathbb{R}^2$ two sets of detected keypoints in image I^1 and I^2. Then the region around each keypoint is transformed into *descriptor space* $\mathbb{R}^{128}$. Therefore there are two sets of descriptors $D^1 = \{\mathbf{d}_1^1, \mathbf{d}_2^1, ..., \mathbf{d}_M^1\}$ and $D^2 = \{\mathbf{d}_1^2, \mathbf{d}_2^2, ..., \mathbf{d}_N^2\}$ with $\mathbf{d}_i^k \in \mathbb{R}^{128}$.

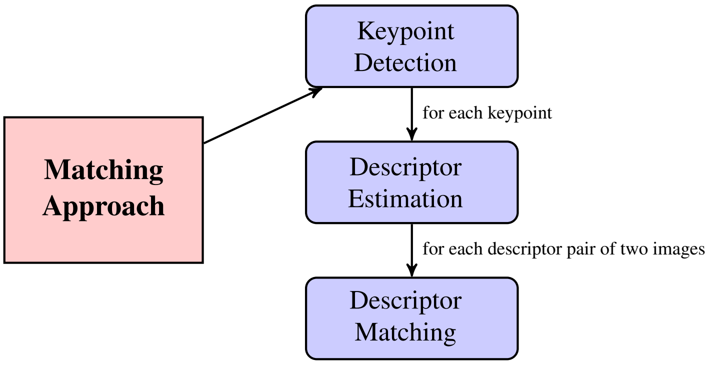


Figure 3.1: Matching pipeline. Included steps are keypoint detection, descriptor estimation and the final descriptor matching.

A matching is a set $M = \left\{(\mathbf{d}_i^1, \mathbf{d}_j^2) \mid \mathbf{d}_i^1 \in D^1, \mathbf{d}_j^2 \in D^2\right\}$. The descriptor space is also called *feature space* and a descriptor is also called *feature vector*. A descriptor is a numerical representation of the surrounding region of a keypoint in an image. It is postulated that the distance under a certain metric, e.g. Euclidean norm:

$$dist(\mathbf{d}^1, \mathbf{d}^2) = \sqrt{\sum_{i=1}^{n} (d_i^1 - d_i^2)^2}, \tag{3.1}$$

is correlating with the similarity of the underlying region. For similar regions the distance between the descriptors should be lower than for dissimilar regions. That enables the matching of corresponding regions, i.e. keypoints, by searching for the nearest neighbor in the descriptor space.

The matching pipeline is not only valid for SIFT, but also for sRD-SIFT and SIFT Sphere. In the following sections each of the stages keypoint detection, descriptor estimation and descriptor matching will be explained, for the three approaches, in detail.

3.2 Keypoint Detection

In the case of SIFT [29] the detection of keypoints in an image includes several stages , as shown in Figure 3.2. All these general steps are also the same for sRD-SIFT and SIFT Sphere. To detect scale invariant interest points, the so called *scale space* of an image is calculated. This is a representation of the different levels of details in an image. A similar effect can be achieved by resizing an image. Since the scale space representation depends on the image structure, it is influenced by radial image distortion, in particular by omnidirectional images. In the following subsections it is shown how sRD-SIFT and SIFT Sphere are adapted for the scale space estimation.

In the next step of the keypoint detection pipeline, the so called *Difference of Gaussians (DoG)* is calculated. These are the subtractions of adjacent layers from the scale space. The

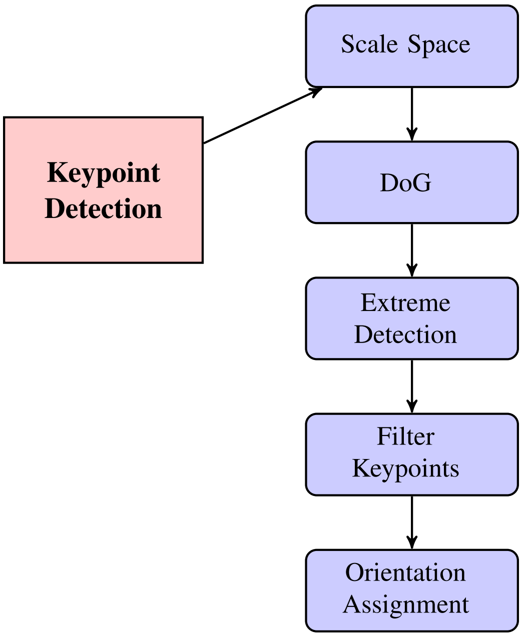


Figure 3.2: Keypoint detection pipeline of SIFT based approaches.

detection of extrema locations of the DoG is followed. A location is an extrema if all surrounding pixel values have a smaller or larger value than the respective position. All extrema are potential keypoints. To remove unstable keypoints in the step *Filter Keypoints*, those extrema which have a contrast below a certain threshold or are lying on an edge and have therefore an unstable location, are removed. This stage of the pipeline also includes the fitting of a certain function on the extrema to interpolate its location and reach, hence a more accurate location. In the final step, an orientation, based on the image gradients is assigned to each extrema that makes the keypoints invariant to image rotations.

Since the keypoints are detected in scale space, each keypoint corresponds to a specific scale value, which is measured in pixels. The scale value represents the size of the underlying image structure from which the keypoint is recognized. For the calculation of the descriptor it is taken to define the size of the underlying region. In the following figures, keypoints are always shown as a circle with a radius according to its scale value and a line, originating from its center, pointing in the direction of the keypoint's orientation.

sRD-SIFT and SIFT Sphere propose new methods for estimating the scale space of an omnidirectional image. All other steps of the pipeline are not affected and only have slight underlying changes. The scale space is the only affected part in the keypoint detection in the case of omnidirectional images [28]. All other stages are relying on the scale space, but do not apply any global image transformation which can be influenced by radial distortion. In Figure 3.3 keypoints detection for SIFT, sRD-SIFT and SIFT Sphere are shown respective their image domain.

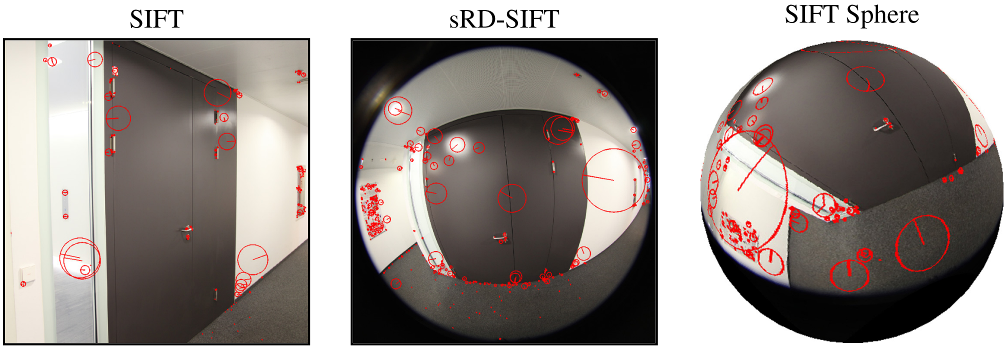


Figure 3.3: Keypoints detected by SIFT, sRD-SIFT and SIFT Sphere in each respective image domain.

SIFT

The stages of the keypoint detection pipeline for SIFT, as shown in Figure 3.2, are composed in detail as follows. The scale space is obtained by transforming the image with the use of a cascading filtering approach, as proposed by Lowe [29]. This estimation is based on the scale space representation of signals introduced by Witkin [47]. For planar perspective images the scale space is defined as

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{3.2}$$

where x and y are image coordinates, I is the input image, $*$ is the convolution operator and G is

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2\sigma^2}(x^2+y^2)}. \tag{3.3}$$

which is the Gaussian function with variance σ^2. This parameter represents the actual scale value in the scale space. As $L(x, y, \sigma)$ is discrete in all parameters, the scale space representation is a stack of increasing blurred images from the input (see Figure 3.4). From this definition it can be observed that, it is only valid in case of non-distorted images. The reason is that the convolution with the Gaussian function requires spatial shift invariance, otherwise the image is not constantly blurred. For distorted images this premise is not fulfilled because image content is deformed.

Lowe also propose an efficient method for calculating the scale space representation of an image [29]. The optimization plays an important role to understand the parameter of SIFT implementations involved. The scale space, as shown in Figure 3.4, is divided into multiple octaves. To produce the images of the first octave the original image is incrementally convoluted with Gaussians having the initial value σ_0 and a constant factor k. To complete an octave if $2\sigma_0$ is reached, Lowe chooses $k = 2^{(1/s)}$, where s is the amount of intervals inside one octave. After finishing one octave, the last Gaussian image that has $2\sigma_0$ is resampled by taking every second pixel in each row and column. The resampled image is taken as the initial layer of the next octave and then is only half of its previous size. Therefore the next Gaussian convolution kernel also has to be the half size of the previous. Therefore the next octave starts with σ_0 again. This is called sampling relative to σ [29].

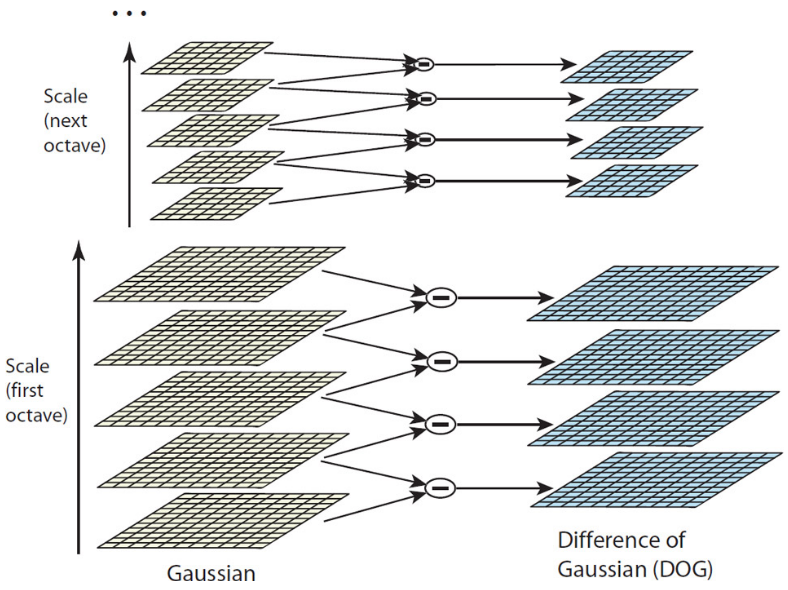


Figure 3.4: Scale space of an image (left) and the corresponding Difference of Gaussians (right). Different octaves in the sale space are obtained by resampling the image to the half size. And different layers in one octave are calculated by consecutive Gaussian convolution. Image is taken from [29].

For detecting stable keypoint locations, Lowe introduces the *Difference of Gaussian (DoG)* as

$$DoG(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{3.4}$$

with a constant factor k, to separate two adjacent scales [29]. DoG is a close approximation of the Laplacian of Gaussians, proposed by Lindeberg [27]. Stable keypoints are detected as local extrema of the $DoG(x, y, \sigma)$. Therefore each pixel is compared to its eight neighbors of the current DoG image with σ and to its nine neighbors in the scale below with $(k-1)\sigma$ and above with $(k+1)\sigma$. The pixel is only taken as a keypoint candidate if its value is larger or smaller than to all of the compared values.

In this step the detected keypoint locations are at exact pixel and scale positions in the scale space. This is only a quantized representation of the captured scene and the optimal extrema of the DoG probably does not lie directly on this sampled position. Therefore Lowe uses a method developed by Brown [5] to determine the optimal position. It fits a 3D quadratic function to the local keypoint position and estimates an interpolated location of the extremum. To approximate the underlying continuous signal he uses the Taylor expansion (up to the quadratic terms) of the scale space function DoG

$$DoG(\mathbf{x}) = DoG + \frac{\partial DoG^T}{\partial \mathbf{x}} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 DoG}{\partial \mathbf{x}^2}\mathbf{x} \tag{3.5}$$

with $\mathbf{x} = (x, y, \sigma)$, for details see [29]. This processing step is combined with the estimation of stable keypoints in the keypoint detection pipeline (see Figure 3.2) of the stage *Filter Keypoints*.

Extrema with low contrast are rejected to finally obtain only the keypoints which are stable against image transformations. For stability, keypoints lying on edges are also eliminated. These are not robust for small amounts of noise [29].

So far keypoints with a specific scale and location are detected. To make them also invariant to image rotations an orientation is assigned to each keypoint. This calculation is based on the local image gradients. The image with the respective scale of each keypoint is taken from scale space, and a histogram from the gradients of the keypoint's regions is formed. The gradient magnitude

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \tag{3.6}$$

and the orientation

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1))/(L(x+1, y) - L(x-1, y))) \tag{3.7}$$

is computed for each location (x, y) in L and added to the respective histogram. The orientation of a keypoint is then set to the orientation of the highest peak in the histogram [29]. For several peaks with a maximal relative difference of 20% multiple keypoints are generated.

sRD-SIFT

Keypoint detection with sRD-SIFT consists of the same principles used for SIFT. Scale invariant keypoints are detected in scale space as extrema of the DoG and an orientation is assigned through the estimation of local gradients. Optimization of the keypoint location and stability is applied in the same way as for SIFT. The difference between SIFT and sRD-SIFT is the estimation of the scale space, because this processing step is affected by omnidirectional images. For sRD-SIFT, it is assumed that the image can be undistorted by a proper model to remove radial distortion. For omnidirectional images this is only partially valid, since the entire view of more than 180 degrees cannot be uniquely mapped to a plane. Nevertheless, the inner part of the image can still be described by such a model [21].

Before the algorithm for estimating the scale space is discussed in detail, the distortion model used is described. Lourenco et al. assume that the image distortion follows the *first-order division model* proposed by Fitzgibbon [28], [12]. Hence the amount of distortion depends only on a single parameter ξ and the center of distortion is approximated by the image center [28]. The pixel displacement between a distorted and undistorted image is in the case of the division model

$$\mathbf{x} = f(\mathbf{u}) = \begin{pmatrix} f_x(\mathbf{u}) \\ f_y(\mathbf{u}) \end{pmatrix} = \begin{pmatrix} \frac{2u}{1+\sqrt{1-4\xi(u^2+v^2)}} \\ \frac{2v}{1+\sqrt{1-4\xi(u^2+v^2}} \end{pmatrix} \tag{3.8}$$

whereby $\mathbf{x} = (x, y)^T$ is a point in the distorted image and $\mathbf{u} = (u, v)^T$ refers to the corresponding point in the undistorted image. Equation 3.8 describes how to apply distortion on a distortion-free image. One of the main benefits of the division model is that there is a simple explicit inversion

$$\mathbf{u} = f^{-1}(\mathbf{x}) = \begin{pmatrix} f_u^{-1}(\mathbf{x}) \\ f_v^{-1}(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \frac{x}{1+\xi(x^2+y^2)} \\ \frac{y}{1+\xi(x^2+y^2} \end{pmatrix}. \tag{3.9}$$

Since the amount of distortion is quantified by a single parameter, Lourenco et al. propose to provide the distortion in relative terms as

$$\%_{distortion} = \frac{r_M^u - r_M}{r_M^u} \times 100 \tag{3.10}$$

with radius $r = \sqrt{x^2 + y^2}$ [28]. r^u is the radius of undistorted points and r_M is the distance from the image center to the corner, which is for a given image the maximal distorted radius.

Using image resampling, the scale space is estimated by the convolution of the Gaussian function (see above) on an undistorted version of the original image. With this in mind, Lourenco et al. formulate an algorithm that estimates the equivalent result, what is obtained if first, the original image is undistorted, then smoothed with a Gaussian kernel and then distorted again to the original image domain [28]. The convolution operation of an undistorted image I^u is

$$L^u(s,t;\sigma) = \sum_{u=-\infty}^{+\infty} \sum_{v=-\infty}^{+\infty} I^u(u,v)G(s-u,t-v;\sigma) \tag{3.11}$$

where G is the Gaussian function as defined in Section 3.2 and σ is the standard deviation of G. From 3.8, 3.9 and 3.11 Lourenco et al. deduce an explicit solution of an adaptive Gaussian convolution on distorted images [28], which is

$$L(h,k;\sigma) = \sum_{u=-\alpha}^{\alpha} \sum_{v=-\alpha}^{\alpha} I(x,y)G\left(\frac{h - x + \xi r^2(h\delta^2 - x)}{1 + \xi r^2(1 + \delta^2 + \xi r^2\delta^2)}, \frac{k - y + \xi r^2(k\delta^2 - y)}{1 + \xi r^2(1 + \delta^2 + \xi r^2\delta^2)}; \sigma\right) \tag{3.12}$$

with (h,k) being the distorted counterparts of (s,t) and with

$$\delta = \frac{d}{r} = \frac{\sqrt{x^2 + y^2}}{\sqrt{h^2 + k^2}}, \alpha = \frac{1}{\sqrt{-\xi}} \tag{3.13}$$

where δ is the ratio of the two radii d and r. For the entire deduction see [28]. Without resampling the image, from Equation 3.13 the scale space of distorted images can be estimated directly. The major disadvantage is that the filter function of the convolution varies with the image location that is being filtered. Thus this operation is called *adaptive convolution*. The drawback is a drastic increase of computational runtime, because the kernel has to be recalculated at each location [28]. Lourenco et al. propose an approximation of this convolution for which the filter kernel only changes for different image radii. Additionally, the convolution is separable, which means that the convolution can be done in x and y-direction separately. With these optimizations the runtime performance of SIFT is reached [28].

To finally estimate the keypoints from the adapted scale space, all other steps of the keypoint detection pipeline (see Figure 3.2) are executed in the same ways as for SIFT.

SIFT Sphere

In Chapter 2 it is shown that all central omnidirectional images can be mapped uniquely onto the sphere. In the spherical image domain the image is not affected by image distortion and the

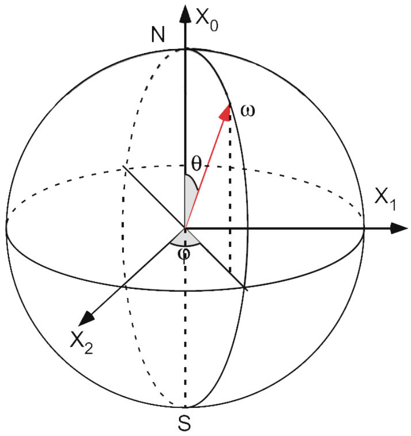


Figure 3.5: Unit sphere with spherical coordinate system. θ is the angle around the x_1 axis and φ is the angle around the x_0 axis. Image is taken from [8].

resolution is equiangular. Therefore Cruz-Mota et al. propose to estimate the scale space for computing SIFT keypoints on the sphere [8]. Thus, the image domain is changed from planar to spherical and SIFT Sphere can be estimated for all central omnidirectional images, which are previously mapped onto the sphere. As a consequence, all subsequent processing steps to estimate local keypoints have to be adapted to the spherical domain.

Mapping an omnidirectional image onto the sphere can only be achieved with a proper model for the specific omnidirectional image. The model used in this work to evaluate the images is described in Sec 4.1. The mapping is performed on the unit 2-sphere by

$$f : \mathbb{R}^2 \rightarrow S^2 \tag{3.14}$$

$$(\theta, \varphi) = f(x, y) \tag{3.15}$$

with $\theta \in [0, \pi)$ and $\varphi \in (0, 2\pi]$. (x, y) is a point in local image coordinates. θ and φ are the two angles as shown in Figure 3.5, which uniquely describe each point on the unit 2-sphere. θ is the polar angle (rotation around x_1) and φ the azimuthal angle (rotation around x_0). If an image is projected onto the sphere, then the resulting pixels refer to angular coordinates. Such an image is shown in Figure 2.3 (omnidirectional spherical). The vertical axis corresponds to θ from 0 to π degrees and the horizontal axis corresponds to φ from 0 to 2π degrees.

The scale space estimation in spherical coordinates requires convolution in spherical coordinates. Cruz-Mota et al. state, that this is hard to compute and propose to calculate the spherical convolution in the spherical Fourier domain, which is obtained by the spherical Fourier transform [8]. If the convolution is directly performed in the omnidirectional spherical image domain or in the omnidirectional planar image domain, the results obtained are incorrect as shown in Figure 3.6. This experiment is implemented by Bülow et al. [6]. The leftmost image is a synthetic texture mapped to the sphere. Figure 3.6 (b) is obtained by Gaussian smoothing in the omnidirectional planar domain. In this case, the image center is more affected by blur than

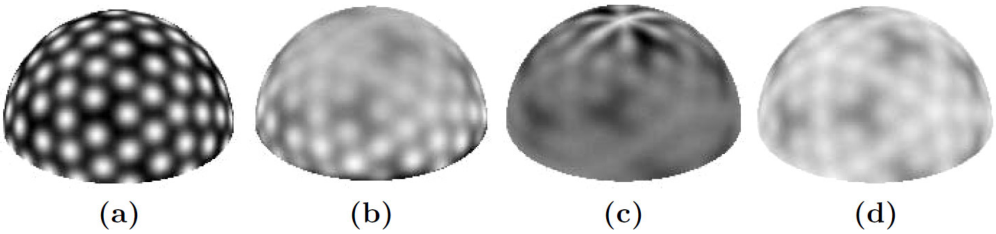


Figure 3.6: (a) A synthetic generated image on the upper hemisphere. (b) After Gaussian smoothing in the omnidirectional planar image domain. (b) After Gaussian smoothing in the omnidirectional spherical domain and (d) after spherical Gaussian smoothing within the spherical Fourier domain. Image is taken from [6].

being close to the equator. An opposite effect is shown in Figure 3.6 (c). Here the Gaussian smoothing is performed in the omnidirectional spherical domain. Only Figure 3.6 (d) contains a correct Gaussian smoothing, which is obtained by spherical smoothing in the spherical Fourier domain.

Let $f, h \in L^2(S^2)$ be two functions defined on the 2-sphere $S^2 \in \mathbb{R}^3$. The spherical convolution is then

$$(f * h)(\omega) = \int_{r \in SO(3)} f(r\eta)h(r^{-1}\omega)dr \tag{3.16}$$

with $\omega = (\theta, \varphi) \in S^2$. The convolution can be expressed as the point-wise product of their spherical Fourier transform

$$\widehat{(f * h)}(l, m) = 2\pi\sqrt{\frac{4\pi}{2l+1}}\widehat{f}(l, m)\widehat{h}(l, 0), \tag{3.17}$$

where $\widehat{(\cdot)}$ is the spherical Fourier transform of a function. For details on how to calculate the spherical Fourier transform see [8].

The major drawback of the spherical Fourier transform is the bandwidth limitation. The bandwidth is the actual sampling rate of the spherical image. For increased values the computation time increases drastically. For an image with a resolution of 768 × 786 pixels it takes 5:42 minutes and for 1280 × 1280 pixels it takes 26:40 minutes on an Intel Core i7-620M CPU at 2x 2.66 GHz with the provided implementation from [8].

Cruz-Mota et al. estimate the scale space by consecutive spherical convolution and downsampling of the spherical image [8], similar to the scale space estimation of SIFT. The same number of octaves and inter-stages of an octave are used. The processing steps for estimating local keypoints are the same as for SIFT, but with the difference that they take place in the spherical domain. An important difference is that the image has no real borders since it represents the surface of a sphere. Another difference during the keypoint calculation is that in the step *Filter Keypoints* the estimation of an accurate extrema location only takes place in the spatial domain and not also in the scale domain [8]. The final orientation assignment is performed using the gradients, which are also computed in the spherical image domain.

3.3 Keypoint Descriptor

The keypoint descriptor estimation pipeline contains the following steps as shown in Figure 3.7. Since the general idea of the SIFT descriptor is to generate a histogram over the local gradients of the region around the keypoint, the first step is to estimate the gradients in that region. All of the following calculations have to be completed for each of the detected keypoints separately. The next step is a Gaussian weighting of the estimated gradients. From that, 4×4 gradient histograms with 8 bins are generated. The resulting descriptor is a 128-dimensional vector, which contains all bins from all histogram. The final step is the normalization of the descriptor vectors to length 1.0, to reduce the effects of illumination changes [29].

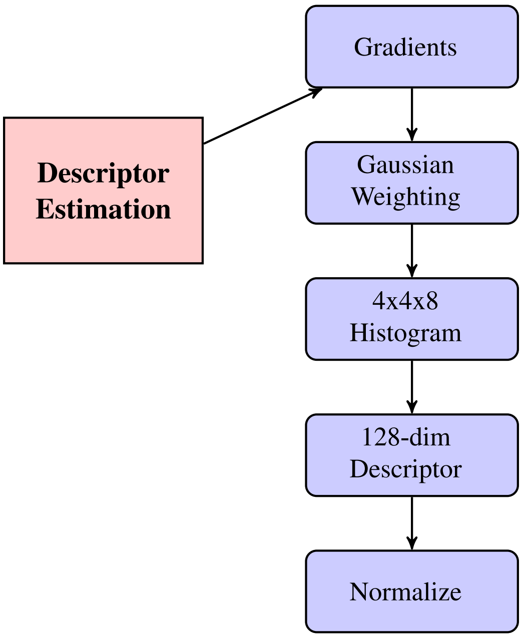


Figure 3.7: Descriptor estimation pipeline of SIFT based approaches.

Mikolajczyk et al. propose that descriptors should be distinctive and at the same time robust to changes in viewing conditions [31]. In case of SIFT, distinctiveness is achieved by using a high dimensional, i.e. 128-dimensional, descriptor vector. Hence the descriptor of a region around a keypoint can be much more precise than with an e.g. 10 dimensional descriptor. On the other hand, a too specific region description may not be robust against image transformations, which is claimed at the same time. SIFT reaches invariance to several transformations, i.e. scale, image rotation, viewpoint changes up to 50 degrees, by using the scale space representation, estimating a specific orientation for each keypoint and finally using the gradients and not the specific intensity values of each pixel. The quantization of gradient locations and orientations also makes the descriptor robust to small geometric distortions and small errors in the region detection [29].

SIFT

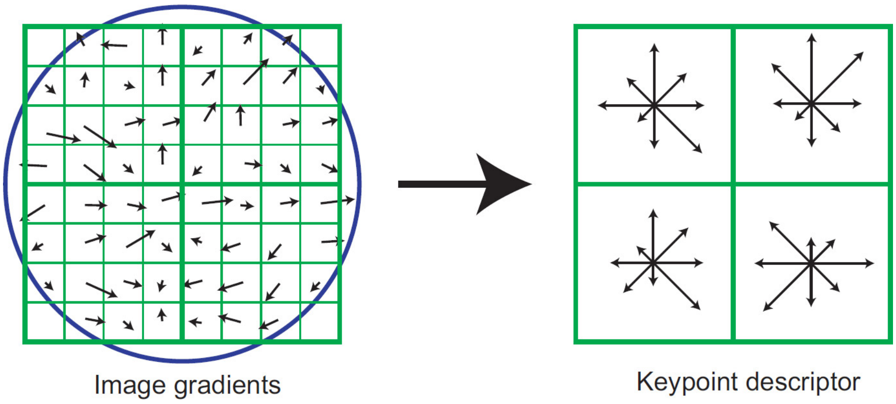


Figure 3.8: Keypoint descriptor estimation of an 8×8 region. The arrow of each sample corresponds to the magnitude and orientation of the underlying gradient. Histograms are formed over 2×2 parts with 8 bins for the gradient orientations. The descriptor is composed from the orientation bins. Image is taken from [29].

In Figure 3.8 the estimation of gradient orientations and magnitudes for an 8×8 set of samples is shown. In contrast to this illustration, SIFT uses 16×16 samples per region. From the gradient orientations and magnitudes, 4×4 histograms are estimated. For simplicity 2×2 are depicted in Figure 3.8. To achieve rotation invariance, before forming the histogram, all gradients and the coordinates of the descriptor are rotated relative to the keypoint orientation [29]. The gradients are computed on the scale space image with the specific scale of the keypoint. This makes the descriptor invariant to scale changes.

Additionally a Gaussian weighting is applied before forming the gradient histograms. In Figure 3.8 this is shown with a blue circle. A weight from that is assigned to the magnitude of each sample point. The Gaussian function ensures that the weight falls off smoothly [29]. Hence, small spatial changes of the region from which the descriptor is calculated do not cause sudden changes of the descriptor.

For each orientation histogram, the gradient orientations are sampled into 8 bins. Therefore each bin covers a range of $360/8 = 45$ degrees. The concrete number of samples per region, number of orientation histograms and number of bins for each histogram is experimentally estimated and recommended by Lowe [29] to give the best matching results.

After estimating 4×4 histograms over a region of 16×16 gradient samples, the descriptor is formed from the bins of the histograms. Since each histogram contains 8 bins the dimension of the descriptor is $4 \times 4 \times 8 = 128$.

The final step, for estimating the keypoint descriptor, is the normalization of the 128-dimensional vector. As shown in [29] the normalization is required to obtain invariance to affine changes in illumination.

sRD-SIFT

Lourenco et al. argue that calculation of SIFT descriptors directly on images with radial distortion causes a displacement in descriptor space with respect to the position that it would have in the absence of distortion [28]. The reason is, that non-linear deformation changes the image gradients of the region around the keypoint. As a consequence, Lourenco et al. propose implicit gradient correction to compensate this error in descriptor calculation [28]. The corresponding explicit adaptation would be to warp the image and compute the gradients in the undistorted image.

sRD-SIFT calculates the corrected gradient from the original image domain. This is achieved by

$$I^u(\mathbf{u}) = I(f(\mathbf{u})). \tag{3.18}$$

which follows from Equation 3.8, with the original image I and its undistorted version I^u. Lourenco et al. apply the derivative chain rule to [28] and obtain

$$\nabla I^u = J_f \cdot \nabla I. \tag{3.19}$$

∇I^u are the gradients in the undistorted image and ∇I corresponds to the gradients in I. J_f is the Jacobian matrix of function f, i.e. the division model. In [28] the Jacobian matrix is deduced from Equation 3.9 in terms of distorted image coordinates $\mathbf{x} = (x, y)^T$ as

$$J_f = \frac{1 + \xi r^2}{1 - \xi r^2} \begin{pmatrix} 1 - \xi(r^2 - 8x^2) & 8\xi xy \\ 8\xi xy & 1 - \xi(r^2 - 8y^2) \end{pmatrix} \tag{3.20}$$

with radius r of $\mathbf{x}$.

From the corrected gradients, histograms are estimated in the same way as for SIFT. Also, the final descriptor is calculated in the same way with one exception, the Gaussian weighting function $G(x, y; \sigma)$ is adapted for the non-linear distortion with $G(x, y; (1 + \xi r^2)\sigma)$. The resulting descriptors from sRD-SIFT are therefore invariant to image distortion described by the division model.

SIFT Sphere

For SIFT Sphere, two different descriptors are proposed [8]. LSD has the purpose of matching omnidirectional images with each other and LPD is proposed to match omnidirectional images with perspective images. Before explaining the specific algorithm for the calculation of the descriptors, first an example of differently estimated scale space regions is given, from which LSD and LPD are obtained.

In Figure 3.9 three different images of the same corresponding region from one keypoint are shown. The LSD is computed on the content of the first image. For matching perspective images, this patch is first projected onto the tangential plane. The resampled result is the second image. From this the LPD is calculated. To finally match this region against a corresponding region of a perspective image, a SIFT descriptor of the perspective corresponding region has to be computed. The third image is the underlying region of the last step. In the optimal case the second and third image patches should look the same, expect for affine transformations.

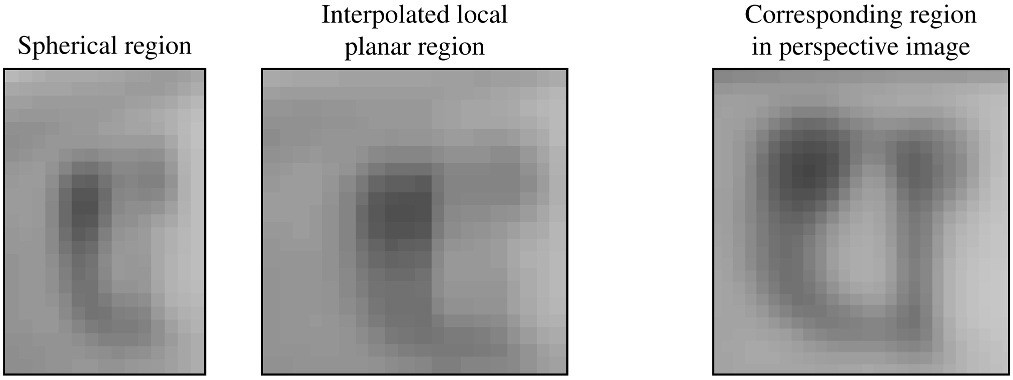

Spherical region
Interpolated local planar region
Corresponding region in perspective image

Figure 3.9: Region of a keypoint from spherical octave image of spherical scale space (left). Projected and interpolated local planar region from spherical octave image (middle) and the corresponding planar region from a corresponding perspective image (right).

Actually, the second image is highly affected by the interpolation. The first image does not have the same resolution as the third image, although they correspond to the same scale value. The descriptor matching of LPD and standard SIFT descriptor finally fails in this case. Specific descriptor matching performance is shown in Section 5.2.

The major different of SIFT and SIFT Sphere is the underlying image domain. Since SIFT Sphere estimates keypoints in the spherical image domain, the descriptor calculation has to be adapted. This is done by formulating a proper method for estimating the gradients. Using spherical gradients, the stages from the descriptor estimation pipeline (see Figure 3.3) can remain the same [8].

Cruz-Mota et al. propose to estimate the orientation of a point $(\theta, \varphi) \in S^2$ with scale σ in spherical scale space L^{S^2} as [8]

$$\alpha(\theta, \varphi, \sigma) = \arctan\left(\frac{L_{\varphi}^{S^2}(\theta, \varphi, \sigma)}{L_{\theta}^{S^2}(\theta, \varphi, \sigma)}\right). \tag{3.21}$$

Equally an adaptation of the magnitude of the gradient at that point (θ, φ) is proposed as

$$m(\theta, \varphi, \sigma) = \sqrt{L_{\varphi}^{S^2}(\theta, \varphi, \sigma)^2 + L_{\theta}^{S^2}(\theta, \varphi, \sigma)^2}. \tag{3.22}$$

From the adapted orientation and magnitude of each gradient, histograms are formed from a $3\sigma \times 3\sigma$ region around the respective keypoint. Through normalization the final descriptor is calculated similar to SIFT.

Because LSD depends on gradients of the spherical image domain it is not compatible with matching SIFT descriptors from planar image gradients. Therefore the LPD is an adaption, to make the descriptor matchable with standard SIFT descriptors of planar perspective images. The region around each keypoint p_i of $L^{S^2}(\theta, \varphi, \sigma_i)$ is projected onto the tangent plane in that point through its antipodal point. The resulting region is an approximation of $L(x, y, \sigma)$ in p_i [8]. Since all gradients are different than in the spherical domain, the keypoint orientation has to be re-estimated. The descriptor estimation of SIFT is used to estimate LPD from the projected region, because it is planar and is not affected by non-linear distortion.

3.4 Descriptor Matching

The last step of the matching approach discussed is to associate descriptors of corresponding keypoints from two images. This is done with a nearest neighbor search in the descriptor space. In other words for each descriptor of one image the closest descriptor in descriptor space is searched for. Depending on the matching strategy, the descriptor is either taken as a match or not.

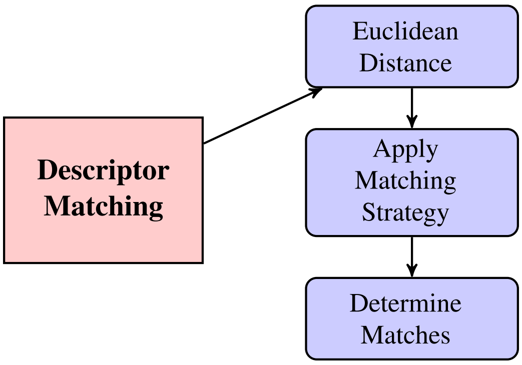


Figure 3.10: Descriptor matching approach.

The descriptor matching pipeline is shown in Figure 3.10. It is the same for matching descriptors of SIFT, sRD-SIFT or SIFT Sphere. In the first step of finding corresponding keypoints, the Euclidean distance is pair-wise calculated for all descriptors of two images.

The next step includes the nearest neighbor search. It depends on the matching strategy. A simple strategy is to associate each descriptor of one image with the closest descriptor of the other image in the descriptor space. However, all keypoints for which no corresponding keypoint exist are falsely matched. Therefore, another matching strategy is to take the nearest neighbor only if the distance lies under a certain threshold. If the application requires finding as many correct matches as possible, by accepting additional false matches, then all descriptors can be taken as a match under a certain threshold. Contrary to this, if false matches are to be avoided, then a better matching strategy is to associate a descriptor only if the ratio between the closest and the second closest descriptor is lower than a threshold. If this is true the match is assumed to be correct, otherwise no match is assigned to the keypoint. The strategy prevents the case where similar but not corresponding descriptors are mistakenly matched. Different matching strategies are evaluated in Section 5.2.

3.5 Summary

This chapter introduced the matching approach of SIFT and its adaptations for omnidirectional images, sRD-SIFT and SIFT Sphere. The keypoint detection and descriptor matching were presented separately. It was shown that the general matching pipeline is the same for each approach. Only scale space estimation has to be adapted for keypoint detection in non-linear distorted images. sRD-SIFT achieves this through adaptive Gaussian convolution with the division model. In contrast, SIFT Sphere warps the planar image onto the spherical surface and subsequently convolves the image with the spherical Fourier transform. Descriptor calculation requires correction of gradients from scale space images. sRD-SIFT uses again the division model to accomplish this and SIFT Sphere calculates the adapted gradients from the spherical scale space. Finally, this chapter presented the actual descriptor matching, which is for all matching approaches the same.

CHAPTER 4

Experimental Setup

Before going into the final scene evaluation of SIFT, sRD-SIFT and SIFT Sphere on omnidirectional images, an experimental environment has to be set up carefully. This chapter deals with the physical experiment, its setup which includes the different image sequences, the measures chosen for performance estimation and the parameter sets involved of each method. The experimental evaluation of matching perspective vs. omnidirectional and omnidirectional vs. omnidirectional images is based on the elaborated matching evaluation of region detectors [32] and region descriptors [31] for perspective images of Mikolajczyk et al.

The main contribution from the evaluation is the specific behavior of each matching approach related to the appearance of environments captured and to transformations between those images. The performance evaluation enables to determine if a matching approach is invariant to a certain transformation or not and how the degree of dependency is. Since all methods are evaluated in the same environment and with the same measures, their performance can be compared on each scene.

The first section deals with the image data sets, under which circumstances they are recorded and what image types are chosen. Also the ground truth estimation for matching those images is discussed. The next section is about state-of-the-art measures for image matching, which are used for the experimental evaluation. An extensive discussion about parameter sets of each method is given in the last section of this chapter.

4.1 Image Data Sets

The experimental evaluation of SIFT, sRD-SIFT and SIFT Sphere is done with multiple and different image sets. In Figure 4.1 example images of the data sets are shown. The perspective and omnidirectional reference image is included for each sequence. Each sequence contains one perspective and six omnidirectional images of the same scene. In Figure 4.1 only the first, third, fourth and sixth image of each sequence are shown. The perspective and the first omnidirectional image are used as the reference image and are then matched against all other five

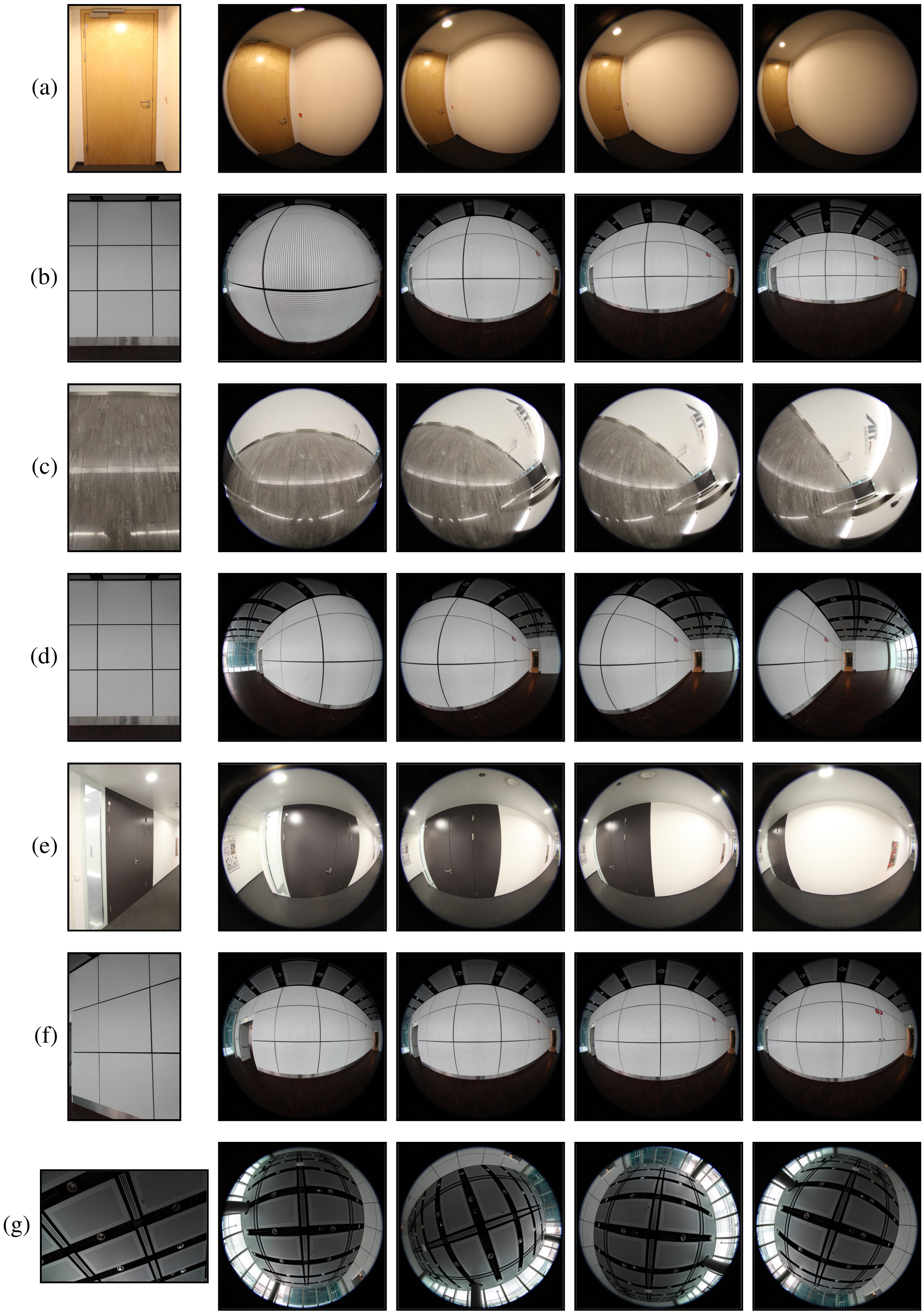


Figure 4.1: Data set used for the evaluation. For each sequence (a-g) the perspective image and the first, third, fourth and sixth omnidirectional image is shown. (a) (door), (b) (wall$_1$) Scale change, (c) (floor), (d) (wall$_2$) field of view change, (e) (entrance), (f) (wall$_3$) viewing angle change and (g) (ceiling) rotation change.

omnidirectional images of each sequence in the experimental evaluation. All images are captured indoor and contain walls, doors, ceilings and floors. These are all surfaces on planes and therefore accomplish the ground truth estimation requirement. As in Fig 4.1 all scenes are not rich in image structure. In several cases the dominant elements are straight lines. The overall structure is composed of homogenous and uniform regions.

The following different imaging conditions are evaluated: scale changes (a) & (b), field of view changes (c) & (d), viewing angle changes (e) & (f) and rotation changes (g). Similar to Mikolajczyk et al. [32] two different scene types are evaluated for all transformations except for the rotation changes. Specifically these are structured and textured scenes. The structured type contains homogeneous regions with non repetitive content and arbitrary shapes, e.g. a wooden door. The textured scenes include images with repetitive patterns, e.g. a wall with squared panels. The distinction to [32] is, that all sequences are captured indoor and therefore contain much less structure or texture. This property influences the detectors in finding less keypoints than in outdoor scenes as in [32]. See Section 4.3 with regards to the actual numbers of detected keypoints in each sequence.

In case of perspective to omni matching, the reference perspective image is always captured from the same viewing angle as the reference omnidirectional image. The perspective image covers the center region of the omnidirectional image in general. It is then possible that for each omnidirectional image of the respective sequence an image region is shared with the perspective image.

Figure 4.2: Sigma AF 8mm 3.5 EX fisheye lens, which is used to capture the omnidirectional images.

The combination of images from planar surfaces and wide angular image acquisition which covers more than an 180° field of view creates the outcome that the entire image can never be covered completely by the planar object. If the camera is placed fronto-parallel to e.g. a wall, then the image borders are not covered by the wall. By rotating the camera around the vertical y-axis, the wall can be projected onto one image border. However, in this case most of the other image parts are not covered by the wall. Hence the image sequences contain both types. In (b), (e), (f) and (g) the images are captured in fronto-parallel view and in (a), (c) and (d) the camera is rotated towards the planar object.

All images are from a Canon 5D Mark II with a Sigma AF 8mm 3.5 EX (see Figure 4.2). The resolution of the perspective images is 1404×934 and of the fisheye images it is 1498×1499.

Image Acquisition

The non-linear properties of the fisheye lenses, results in increasing radial distortion from the image center to its corners. Depending on the location in the image plane, the respective region is distorted differently. Consequently geometric image transformations, such as scaling or viewport changes are no longer linear. Photometric changes e.g. image blur, illumination changes or JPEG compression are not influenced by the omnidirectional vision. These transformations are still equally applied on the entire image surface. Therefore photometric effects are out of scope and not examined. For perspective image matching photometric transformations are already evaluated in [32] and [31].

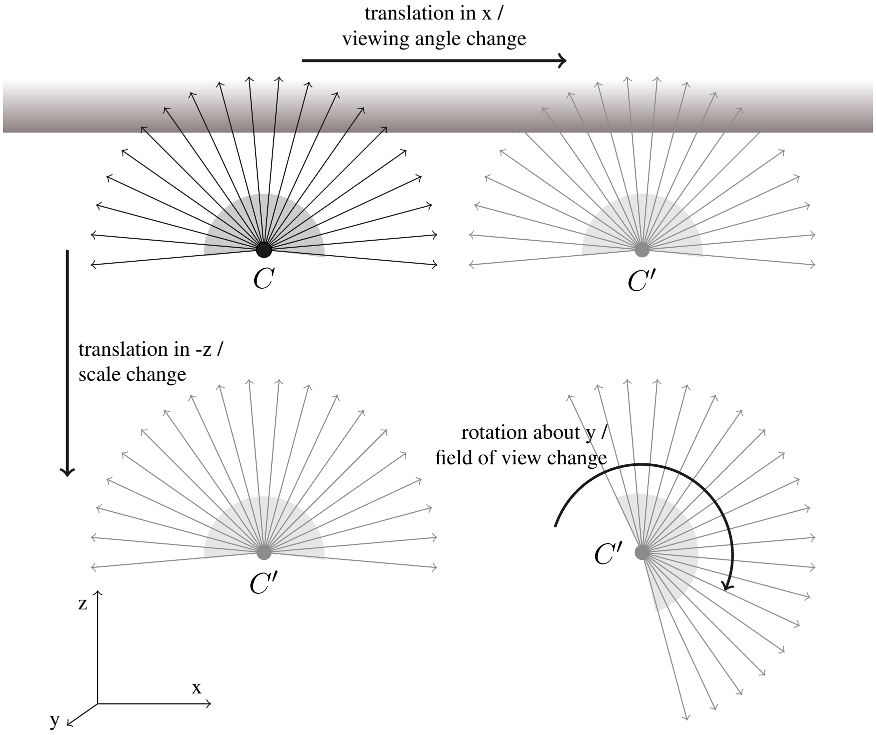


Figure 4.3: Transformations of omnidirectional vision with projection center C related to a planar object. Arrows illustrate viewing rays with $> 180°$ field of view. C' are the transformed projection centers after translating C in x or z direction or rotating C around the y-axis.

In Euclidean 3D space there are 6 possible transformations [19]. For each direction of the 3 dimensions there is a translation and a rotation. In Figure 4.3 the geometric transformations of an omnidirectional vision system are illustrated. The planar object, e.g. a wall is extended in x- and y-direction. Therefore a translation in either the x- or y-direction has the same effect as the distance to the surface remains unaffected. Because translating the viewpoint, this transformation is called viewpoint changes. The last remaining translation, in z-direction, adapts the distance to the plane. The projected image parts increase or decrease in size and therefore this transformation is referred to as scale changes. Rotating the x- or y-axis has again a similar effect, which moves the viewing direction away from the plane. This is called field of view changes. Rotations around the y-axis, which correspond to the camera's viewing direction, are similar to rotating the 2D image around its center and therefore it is named rotation changes.

The experimental evaluation and analysis shows how well each matching approach can deal with a certain geometric transformation regarding different degrees of difficulty. The range of each transformation is measured in degrees of rotation and field of view changes with a scale factor for scale changes. In the case of viewport changes, only the number of the image pair is given, because the relative distance of each viewport is unknown.

Fisheye Camera Model

Since the omnidirectional images are all captured by the Sigma AF 8mm 3.5 EX fisheye lens, a fisheye camera model from Mičušík et al. [30] is used. This model can be applied to radial symmetric omnidirectional images, which implies fisheye images as well.

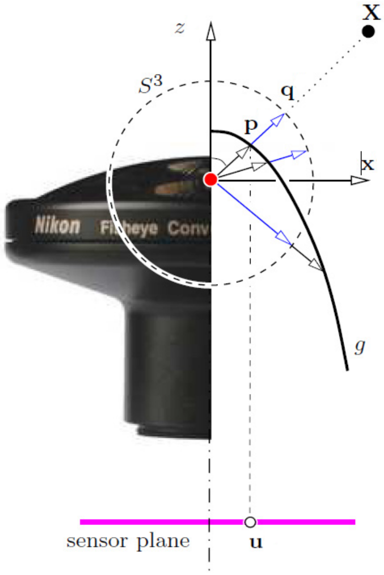


Figure 4.4: Non-linear projection model for wide angular cameras. A point $\mathbf{x}$ is projected onto the sensor plain in $\mathbf{u}$ via the sphere S^3. Image is taken from [30] with slightly changed labels.

Figure 4.4 depicts the general projection procedure. A point $\mathbf{X} \in \mathbb{R}^3$ in 3D space is mapped onto the surface of a unit sphere S^3 in $\mathbf{q}$. S^3 shares its center with the center of projection (red dot in Figure 4.4). $\mathbf{q}$ is in linear projective camera coordinates, since the spherical surface is mapped bijective onto the sensor plane, i.e. each point on the spherical surface has a one-to-one relation to a point in the omnidirectional image. It is also called the camera viewing ray of $\mathbf{u}$.

To cover the non-linear behavior of the fisheye lens, a function $g : \mathbb{R} \times \mathbb{R}^N \rightarrow \mathbb{R}$ is introduced which maps $\|\mathbf{u}\|$ with N model parameters to $\mathbf{p}$. $\mathbf{u} \in \mathbb{R}^2$ is a point in the image plane and $\|\mathbf{u}\|$ corresponds to its radius, i.e. the distance of $\mathbf{u}$ to the image's center of symmetry. $\mathbf{p}$ and $\mathbf{q}$ are 3D vectors with the same direction, but $\mathbf{q}$ has length one, since it is located on the unit sphere. This results in

$$\mathbf{p} = \begin{pmatrix} \mathbf{u} \\ g(\|\mathbf{u}\|, \mathbf{a}) \end{pmatrix}, \text{ with } \mathbf{a} \in \mathbb{R}^N. \tag{4.1}$$

Depending on the actual lens, function g can have different solutions with a different number of parameters $\mathbf{a}$. In general the more parameters involved, the more complex and accurate is the model [30]. Mičušík et al. propose a specific two parametric model for fisheye lenses, like the Sigma AF 8mm 3.5 EX, as

$$g(\|\mathbf{u}\|, a, b) = \frac{\|\mathbf{u}\|}{\tan\theta(a, b)} \tag{4.2}$$

with

$$\theta(a, b) = \frac{1}{b}\sin^{-1}(\frac{b * \|\mathbf{u}\|}{a}) \tag{4.3}$$

and model parameters a, $b \in \mathbb{R}$.

Ground Truth Estimation

A reference measure is needed to evaluate interest point matching. In other words, it must be known if two corresponding points from different images relate to the same 3D point in space. Therefore knowledge about the actual scene and the cameras is required. To simplify this problem, only planar scenes are captured for evaluation. In this case there exists a linear transformation called homography between two images of a planar scene. Let $\mathbf{p_i} = (x_i, y_i, w_i) \in \mathbb{P}^2$ be a 2D point in a reference image whereby w_i is the homogenous part of $\mathbf{p_i}$ and let $\mathbf{p'_i} = (x'_i, y'_i, w'_i) \in \mathbb{P}^2$ be the corresponding point in a second image. If $\mathbf{p_i}$ and $\mathbf{p'_i}$ are located on a plane, then there is a homography $H \in \mathbb{R}^{3\times3}$ with

$$\mathbf{p'_i} = H * \mathbf{p_i}. \tag{4.4}$$

From at least 4 corresponding points on a plane, H can be estimated with the *golden standard algorithm*, for example, described by Hartley and Zisserman in [19].

Similar to the approach in [32] a robust ground truth homography between all image pairs of the test sequences is semi automatically estimated. Since a homography is a linear transformation, the non-linear properties of the omnidirectional images need to be taken into account. An outline of the main six steps of this strategy is given:

(1) Manually select at least 4 points $P = (\mathbf{p}_1, \mathbf{p}_2, .., \mathbf{p}_n)$ in the reference image and corresponding points $P' = (\mathbf{p}'_1, \mathbf{p}'_2, .., \mathbf{p}'_n)$ in the second image as shown in Figure 4.5.

(2) In between the image coordinates of both images, a linear transformation does not exist because of the non-linear distortion of the omnidirectional image. Therefore the image coordinates P' of the non-perspective image need to be transformed to camera coordinates P'_c, which are lying in the linear 2D projective space $\mathbb{P}^2$. This is done by applying the camera model described above in Equation 4.2 and Equation 4.3.

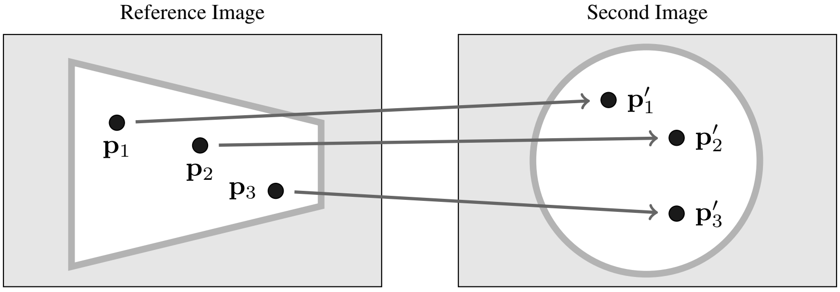


Figure 4.5: Corresponding points in perspective image (left) and omnidirectional image (right).

(3) Estimate an approximate homography H_1 from corresponding point sets P and P'_c with the golden standard algorithm [19].

(4) Warp the second image with H_1 to the reference image as shown in Figure 4.6. Then there remains only a small baseline between the reference image and the warped one. This difference is consistent with the error of H_1.

(5) As the second image is now roughly aligned with the reference image, a small-baseline robust homography estimation algorithm can be applied. Harris Corners [18] are detected and matched by correlation [19]. Then the RANdom SAmple Consensus (RANSAC) algorithm [11] is used to eliminate outliers and retrieve an accurate residual homography H_2. For details see Hartley and Zisserman [19]. The interest point detector for estimating the robust ground truth homography should be independent of all evaluated methods, as proposed by Mikolajczyk et al. [32].

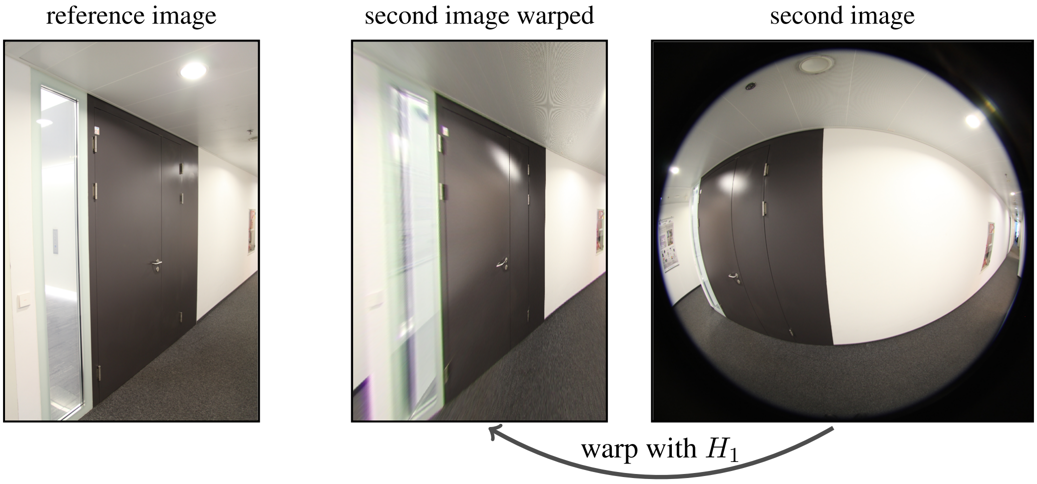


Figure 4.6: Transformation of omnidirectional image (right) to perspective image (middle) by applying the manually estimated rough homography H_1.

(6) Finally estimate H from H_1 and H_2 to receive the robust homography between the

reference image and the second image. Since

$$\mathbf{p}_i' = H_1(H_2\mathbf{p_i}) \tag{4.5}$$

it follows that

$$H = H_1 H_2. \tag{4.6}$$

Note, that in case of estimating H between two omnidirectional images, corresponding points in both images need to be transformed into camera coordinates before applying H.

The mean symmetric transfer error [19] of H_1 from manually set keypoints is 3.33 pixels over all 70 image pairs. This error is minimized to 1.41 pixel, which is the mean symmetric transfer error of H_2 and thus for H. This remaining error results from the accuracy of the omnidirectional image model, since this is involved in the transformation from image to camera coordinates. Another error source is the lens distortion of the perspective image. However, it can be ignored because the error is already small enough to not influence the estimation of the ground truth matches for two images.

4.2 Measures

To compare the performance of different keypoint detectors and their descriptors, performance measures have to be defined. By design each detector detects different locations of interest in one image. Therefore it is not reasonable to test the detection rate of one keypoint over all detectors. As the purpose of an interest point detector is to redetect the same locations in another image, the amount of repeated points can be used to quantify the detector's performance. This measure is called *repeatability*. To evaluate the matching performance of keypoint descriptors, correct matches can be set in relation to false matches.

In both cases of detector and descriptor evaluation, it must be known which location in the reference image belongs to which location in the second image. Due to planarity of the captured image sequences, each point in the reference image can be uniquely transformed to the second image by the manually estimated ground-truth homography and the knowledge of the intrinsic parameters of the fisheye camera.

Subsequently the two measures repeatability [32] and recall-precision [31] are described. The former is used to evaluate the performance of keypoint detectors and the latter is used to evaluate the performance of keypoint descriptors.

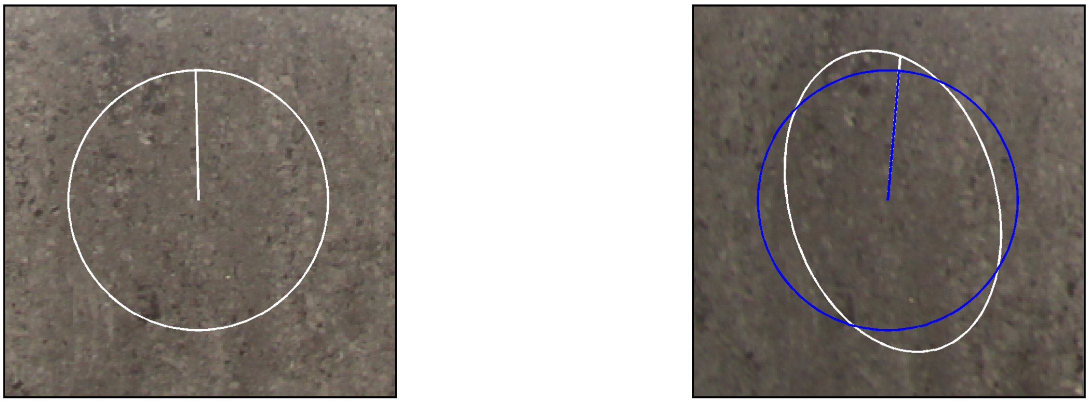
Repeatability

Schmid et al. introduce the evaluation criteria, repeatability, to estimate the performance of interest point detectors and to show their geometrical stability under different transformations for perspective images [40]. The repeatability r_i between a reference image I_{ref} to an image I_i is defined as

$$r_i = \frac{\#K^{true}}{min(\#K_{ref}, \#K_i)} \tag{4.7}$$

where K_{ref}, K_i refers to the set of keypoints detected in image I_{ref}, I_i and $\#K$ is the cardinality of set K. K^{true} is the set of keypoints which are simultaneously detected in image I_{ref}

and image I_i [40]. Since keypoints are, due to sampling errors, not redetected at the exact corresponding position, Schmid et al., define an error range ϵ in which keypoints are still recognized as repeated. The error follows the consistency criterion postulated by M. Lourenço et al. [28]. Another claim is that $0 <= r_i <= 1$ holds. Per definition $r_i = 0$ means that no keypoint from the reference image is redetected and $r_i = 1$ refers to the case that all keypoints in image I_{ref} are redetected in image I_i. In case two points are lying inside the range ϵ only the closest point is taken. Otherwise r_i could become larger than 1. Points, which are not visible in both images or do not satisfy the homography, are excluded from set K [40].



(a) Reference image

(b) Second image

Figure 4.7: Original SIFT feature detected in reference image and its transformed illustration in second image (white). Estimated SIFT feature in second image with adapted scale (blue).

Repeatability measures how many keypoints are matchable with an appropriate descriptor. As the keypoints evaluated not only consist of a location but a scale and an orientation too, the consistency criterion has to be adapted. These additional properties originate from SIFT proposed by D. Lowe. Therefore the following experiments use the same suggested ranges for location, scale and orientation. A detected keypoint k in image I_{ref} is transformed by homography to image I_i with scale σ_{k_i}. The keypoint is only repeated if there is a corresponding point in image I_i, with a distance of less than σ_{k_i} pixel. Its scale value must be in range of $\sqrt{2} * \sigma_{k_i}$ and its orientation can have a maximal difference of 15 degrees [29]. If scale and orientation are not considered, keypoints from the reference image and the second image may be mistakenly assumed to be a match, but actually do not correspond because of a different scale or orientation.

With the homography, it is known where keypoints from the reference image have to be redetected in the second image. As the second image is taken from a different viewpoint the scale and orientation of the keypoint may change too. Therefore these properties need to be adapted. The scale value defines the region used by the descriptor and can be visualized as a circle with its center at the keypoint location [29]. To estimate the altered scale value in the second image, points on the circle are transformed by the homography. The new scale value is set to the mean distance of the transformed circle points to its transformed center, as it is depicted in Figure 4.7. The same idea is used to estimate the transformed orientation. Here only the point with the corresponding angle is mapped to the second image. This heuristic permits to predict the scale and orientation of detected keypoints after their transformation to the second image. Consider that the underlying image domain has to be taken into account, e.g. if the image is represented in spherical coordinates, the keypoint and its scale and orientation have to be projected onto the spherical surface after applying the homography.

The repeatability is evaluated in relative and absolute values. Only both values together are significant, because the relative value can be 100%, for example, with an absolute number of one. Then the descriptor seems to be perfect in relative terms, but practically just one match is unusable. In contrast, a relative value of 10%, for example, with an absolute number of 100 matches, can be still a reasonable result, despite the low relative value.

Recall Precision

The repeatability measure previously shown just evaluates how good keypoints can be redetected under a certain image transformation. Actual descriptor matching is not involved and needs to be evaluated separately. Since there are only correct or incorrect matchable keypoints which can be matched by the approach, a two by two design as shown in Table 4.1 can be establish. Here the four possible cases, *true positives (TP)*, *false positives (FP)*, *true negative (TN)* and *false negative (FN)* are depicted.

	Matchable	Not matchable
Match test positiv	$TP = M^{true}$	$FP = M^{false}$
Match test negative	$FN = K^{true} - M^{true}$	$TN = K^{false} - M^{false}$

Table 4.1: Definitions for sets of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negative (FN). M is the set of all matched keypoints. K is the set of all detected keypoints, whereby K^{true} are matchable keypoints and K^{false} are detected keypoints without a correspondence.

To compare the matching performance of different detectors and their descriptors, the recall versus 1-precision is estimated. This criterion is already used by Mikolajczyk and Schmid [31] to evaluate the performance of local descriptors for perspective images. These measures are defined as

$$recall = \frac{\#M^{true}}{\#K^{true}} = \frac{TP}{TP + FN} \tag{4.8}$$

and

$$1 - precision = 1 - \frac{\#M^{true}}{\#M} = 1 - \frac{TP}{TP + FP} = \frac{FP}{TP + FP}. \tag{4.9}$$

Recall is the percentage of correctly matched keypoints to all actual corresponding keypoints for two images of the same scene. *1-Precision* defines the amount of false matches with respect to the total number of matches. The two measures are evaluated against each other in one graph. The resulting curve corresponds to the matching performance of the respective descriptor. Actual values of precision and recall are estimated with the ground truth matches, which are already used for the detector evaluation (see above). The actual shape of the recall-precision curve results by varying the threshold of the matching.

4.3 Parameter Sets

As SIFT Sphere and sRD-SIFT are developed from the original SIFT [29], they share all parameters from SIFT. Relevant parameters for which the authors of each method suggest different values are σ_0, O_{min} and T_{peak}. The first two values relate to the scale space generation and control the sampling rate in spatial and scale orientation. The third parameter mentioned controls the sensitivity of extrema detection. All other parameters proposed by Lowe [29] stay fixed in sRD-SIFT and SIFT Sphere. For details see [8] and [28].

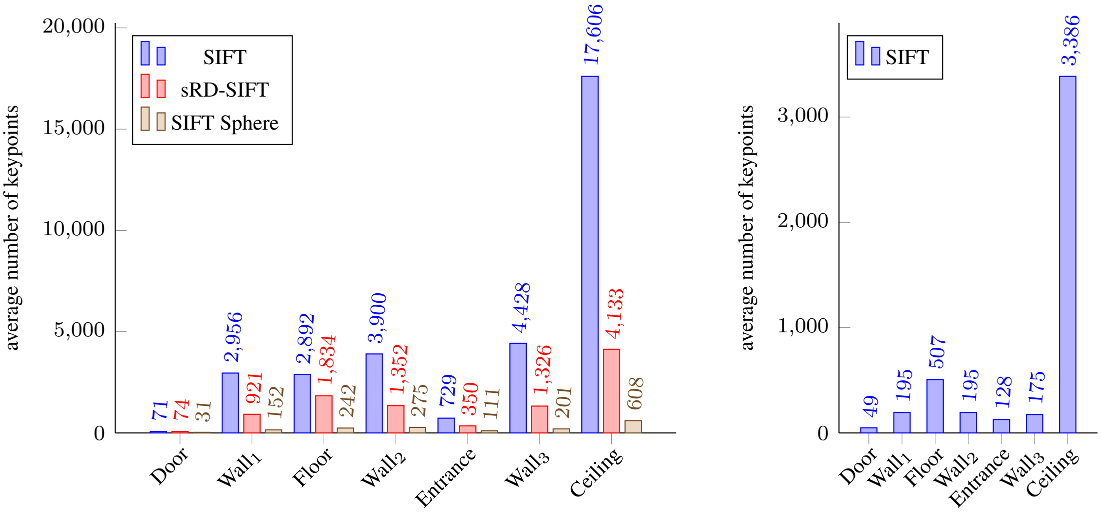


Figure 4.8: Average number of keypoints detected by SIFT, sRD-SIFT and SIFT with original proposed parameter values. Left: omnidirectional images, right: perspective images.

The choice of parameter values influences the keypoint density. If the keypoint density increases, more potential matchable keypoints are detected. Keypoints can also be detected which are not re-detectable under a different viewing angle and thus are not matchable. The number of detected keypoints is not only depending on the parameter values, but on the scene type too. In Figure 4.8 the average number of detected keypoints for each scene is shown. It is important to note here, that the parameter values suggested of each author are used. These are for SIFT $\sigma_0 = 1.6$, $T_{peak} = 0.03$, $O_{min} = -1$ and for sRD-SIFT $\sigma_0 = 1.6$, $T_{peak} = 0.04$, $O_{min} = 0$ and for SIFT Sphere $\sigma_0 = 3.0$, $T_{peak} = 0.02$, $O_{min} = 0$. Lowe claims that in a typical image, several thousand keypoints can be extracted [29]. This assertion stays in strong contrast with the detection numbers in Figure 4.8. Actually, for example, for the door scene the number is below 100 for SIFT. This fact is explained by the application of indoor matching, where the images contain sparse structure and texture. Nevertheless the keypoint density can still

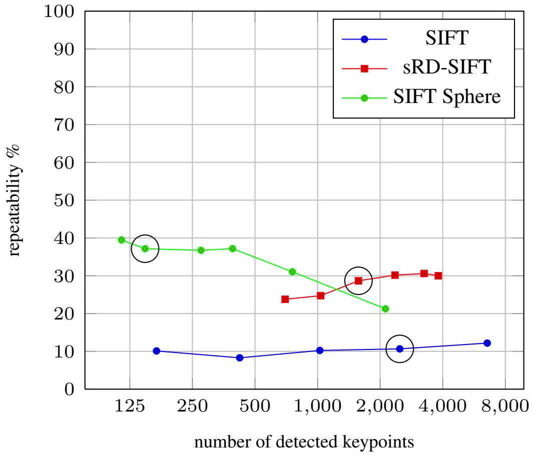


Figure 4.9: Repeatability with respect to number of detected keypoints for matching first and third sample of scene floor. Results for parameter sets suggested from author of each method (black circles).

be increased by changing the parameter values. In perspective images the number of detected keypoints is for SIFT for each scene lower than for the omnidirectional images. The reason is the larger field of view in case of omnidirectional images.

The choice of parameter values not only results in different keypoint densities, but also influences the detection repeatability and the matching performance, as shown by Lowe [29]. In Figure 4.9 this effect is depicted by changing the parameter T_{peak} for SIFT, sRD-SIFT and σ_0 for SIFT Sphere. The reference implementation of SIFT Sphere does not allow the peak threshold to be adapted. For different parameter values, the number of detected keypoints changes drastically, e.g. for SIFT in a range of 168 to 6547, but the effected repeatability varies only in a minor range from 8.3% to 12.1%. In general, the repeatability increases for SIFT and sRD-SIFT for higher keypoint densities and for SIFT Sphere the repeatability decreases from a density of 390 keypoints. The black dots in Figure 4.9 mark the results for the parameter values suggested from the original papers ([29], [28] and [8]). As already shown in Figure 4.8, SIFT Sphere detects much less keypoints with the default parameter values. This is due to the bandwidth limitations of the Spherical Fourier Transformation. Nevertheless, it is possible to increase the number of keypoints by changing parameter σ_0 without losing repeatability score.

The evaluation of the different image sequences is done with the default parameters supplied by the authors, because the rank of the detectors stays the same in the range of the analyzed keypoint densities (see Figure 4.9). One exception is made for SIFT Sphere, because the number of keypoints is low for all scenes in comparison to its competitors. $\sigma_0 = 1.6$ and $O_{min} = -1$, instead of $\sigma_0 = 3.0$ and $O_{min} = 0$ is used. These values are related to 390 keypoints and 37% repeatability for matching the first and the third image of the floor sequence.

One parameter is still not analyzed and may have a significant effect on the matching performance. It is the distortion parameter of sRD-SIFT. The influences of different values for the distortion parameter on the repeatability of sRD-SIFT is shown in Figure 4.10. The range

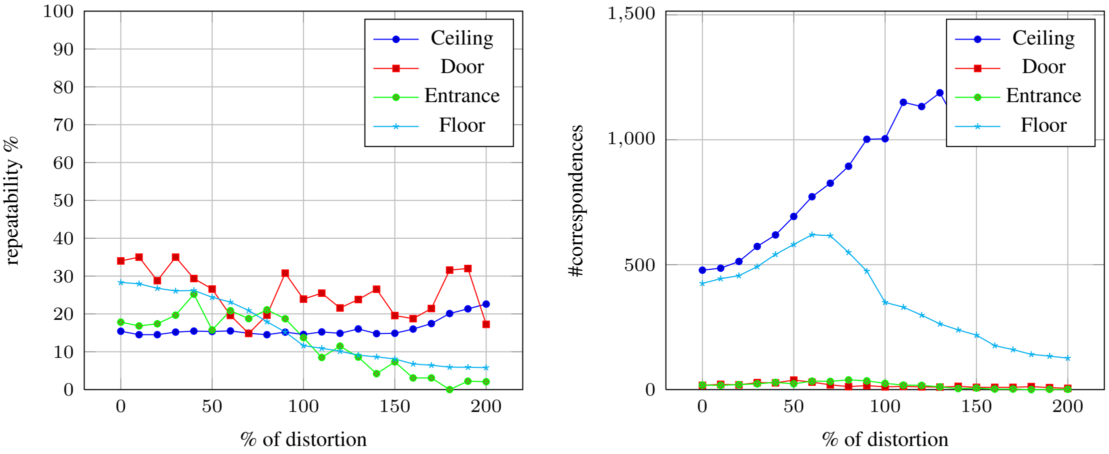


Figure 4.10: Repeatability (left) and number related correspondences (right) of sRD-SIFT with different values of its distortion parameter. The omnidirectional reference image is matched against the third image of the sequences ceiling (g), door (a), entrance (e) and floor (c), see Figure 4.1.

examined is between 0%, which means no distortion and a performance similar to SIFT [28], and 200% distortion (see Equation 3.10). It is clearly observed, that the actual value has a huge influence on the repeatability and the number of correspondences. Also the parameter changes are not invariant to the scene type. The dependency between the distortion value chosen and the scene type follows from the fact that the distortion model of sRD-SIFT cannot describe the distortion of the entire fisheye image correctly. As in each scene type, keypoint density varies in its location, the performance is superior or not, depending on which part of the image is currently described best by the distortion model. Finally, a distortion value of 10% for the scene evaluation is chosen. Larger values of distortion may increase the number of correspondences, but they decrease also the average repeatability score over all scenes examined (see Figure 4.10).

4.4 Summary

This chapter presented all necessary parts for performing the experimental evaluation. First, the image sets are given. Then, image acquisition described the possible camera transformations, which are used to capture the image sets. These are scale changes, rotation changes, field of view changes and viewpoint changes. A fisheye camera model from Micusik et al. was introduced for describing the actual camera used. A major part of this chapter is the explanation of ground truth estimation, since the accuracy of the performance evaluation is depending on that. All substeps of the robust estimation are given. Next, evaluation measures were introduced. Repeatability is used for estimating the performance of keypoint detection and recall vs. 1-precision is the measure to estimate the performance of descriptor matching. This chapter concludes with a discussion of the parameter sets used for each matching approach.

CHAPTER 5

Evaluation

In this chapter the results of the evaluation are presented and discussed. First, the performance of the keypoint detectors from SIFT, sRD-SIFT and SIFT Sphere are compared (see Section 5.1) and estimated on the image data set under different image transformations. These sequences are captured in real indoor environments and not rendered artificially. Using the latter case is also a reasonable source for performance evaluation, since there is already an accurate ground truth available for any image transformation. But as artificially generated data does not model each variable of real cameras, lenses and scenes, an evaluation on natural images gives more realistic results to assess the performance of each matching approach.

The second part of this chapter (Section 5.2) contains the experimental results from the actual image matching. After estimating which keypoint detector renders the most reliable results, here the total overall performance of each approach is examined and discussed. It demonstrates how well each method can convert the performance of the detector into the final matching performance.

In all cases there are not only omnidirectional images matched with each other, but also perspective to omnidirectional matching is evaluated. This can be done since the matching approaches are capable to match both types. For SIFT the VLFeat implementation[1] is used. In the case of sRD-SIFT and SIFT Sphere the implementations provided by the authors Lourenço et al.[2] and Cruz-Mota et al.[3] are taken respectively.

5.1 Results on Keypoint Detection

Matching approaches comprise both a detector and a descriptor. First the detector is evaluated. The descriptor matching can never be better than the keypoint detection, as only redetected keypoints can be matched. Thus the keypoint detector constitutes the basis for any matching approach.

[1] VLFeat: http://www.vlfeat.org/
[2] sRD-SIFT: http://arthronav.isr.uc.pt/~mlourenco/srdsift/
[3] SIFT Sphere: https://sites.google.com/site/javicm/software

Mikolajczyk et al. show that keypoint, specifically region detection, is not invariant to the scene type and any transformation during matching of perspective images [32]. In the current evaluation these properties of keypoint detection are analyzed on omnidirectional images taken in indoor environments. SIFT, sRD-SIFT and SIFT Sphere have to demonstrate if and how good they can redetect keypoints under different geometric transformations, i.e. scale changes, field of view changes, viewpoint changes and rotation changes. One question is how strong the detector performance is actually influenced by the transformation. In addition, similar to the evaluation of Mikolajczyk et al., structured and textured scene types are analyzed for each sequence. In the case of perspective matching, it is already shown that there is no general invariance for each detector. In the following evaluation the role of these two scene types in omnidirectional image matching is shown.

Since SIFT Sphere and sRD-SIFT are designed not only to match omnidirectional images with itself, but to match perspective with omnidirectional images too, this aspect is evaluated under the same conditions as the evaluation of omni to omni matching.

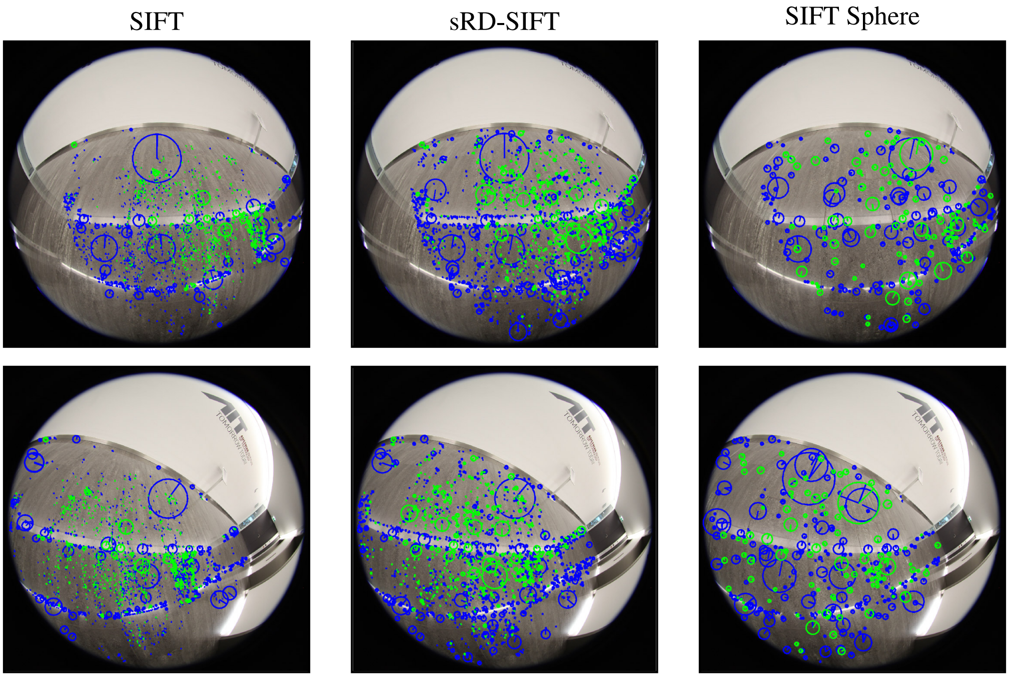


Figure 5.1: Keypoint detections in two different images by SIFT (left), sRD-SIFT (middle) and SIFT Sphere (right). Keypoints in blue are detected only in one image and the green ones have a correspondence in the other image. The circle radius is equal to the scale value and the line the orientation of the keypoint.

Results of the keypoint detections from SIFT, sRD-SIFT and SIFT Sphere are shown in Figure 5.1. Each keypoint is visualized as a circle with a radius equal to its scale value and a line segment starting from the center to the circle border, which indicates the orientation of the respective feature. Keypoints which are redetected in the second image are marked as green and all of the non redetected interest points are colored in blue.

Before discussing the actual evaluation of the different transformations, it can be already observed in Figure 5.1 that SIFT and sRD-SIFT have redetections in the center of the omnidirectional image only. In the case of SIFT, keypoints on average have a small scale of a few pixels. This is due to the fact that SIFT does not consider the omnidirectional, non-linear geometry. In contrast sRD-SIFT obtains keypoints with a larger average scale. SIFT Sphere redetects equally distributed keypoints on the image plane, but despite of interpolation only keypoints with larger scales are recognized as depicted in Figure 5.1.

In the following subsections each transformation is evaluated under the different scene types and image sequences.

Scale Changes

In Figure 5.2 the influence of scale changes for the structured door sequence from Figure 4.1(a) is shown. The repeatability score and the absolute number of correspondences are depicted for omni to omni matching and for perspective to omni matching. The best results for both cases are obtained by SIFT Sphere. This follows from the fact that SIFT Sphere takes the spherical geometry of the fisheye lens into account. Therefore SIFT Sphere can still handle scale changes with a factor 1.8 better than SIFT and even sRD-SIFT.

Door scene	I_{ref}, I_2	I_{ref}, I_3	I_{ref}, I_4	I_{ref}, I_5	I_{ref}, I_6
omni to omni	1.23	1.43	1.57	1.82	2.16
perspective to omni	1.61	2.0	2.25	2.63	3.09
Wall$_1$ scene					
omni to omni	1.3	1.7	2.2	2.7	3.2
perspective to omni	0.8	1.1	1.4	1.7	1.9

Table 5.1: Factors of scale changes between images of the door scene and wall$_1$ scene.

All methods are not invariant to scale changes for the omni to omni matching and for perspective to omni matching. In the latter case, sRD-SIFT is least affected by scale changes. Its repeatability varies between 18% and 28%. Concerning the other approaches, the range starts with 9% to 16% for small scale changes and ends between 21% and 27% for larger scale changes. In the omni to omni matching case the repeatability score of SIFT and sRD-SIFT decreases with a constant difference equally. Together with SIFT Sphere small scale changes (factor 1.2) result in the range of 40% - 45% and end for high scale changes (factor 2.2) at 13% to 18%. These low values are due to the sparse structure of the door scene, which contains small gradients. Therefore the absolute number of correspondences found is very low (5 - 25) in contrast to other scenes with more structure (ceiling sequence Figure 4.1(g)). For omni to omni matching, the largest number of corresponding keypoints is given by SIFT (25) and closely followed by sRD-SIFT (23) and SIFT Sphere (20). Where the maximum given scale change (factor 2.2) is concerned, the number of correspondences decrease in the range of 5 to 7 for all approaches, in perspective to omni matching results in never more than 15 redetections.

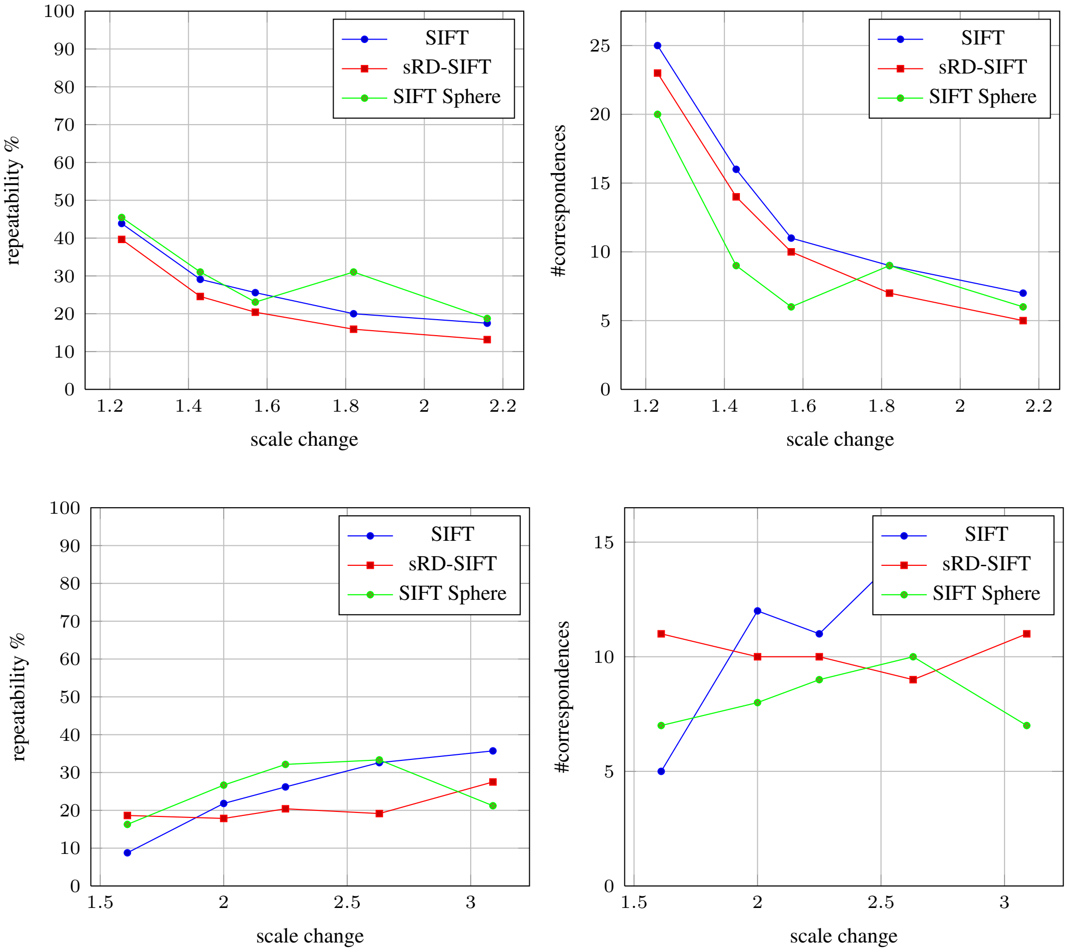


Figure 5.2: Scale changes for structured scene (Door sequence Figure 4.1(a)). (top) Repeatability and number of corresponding points for omni to omni matching. (bottom) Same for perspective to omni matching.

The influences of scale changes on the textured wall$_1$ scene from Figure 4.1(b) are shown in Figure 5.3. Here the results for omni to omni matching are significantly less favorable than in the case of scale changes in the textured scene (Figure 5.2). However SIFT still redetected 107 correspondences, for a scale change of factor 1.3, but this is only 10% of all detected keypoints.

The overall performance lies between 2% and 10% for all approaches, with one distinctive exception of 40% redetected points by sRD-SIFT for scale changes of factor 1.7. This can be explained by the actual scale values of the detected keypoints. Which can correlates with the scale changes of the images and therefore causes this unique exception. Interestingly in the case of omni to omni matching the results are invariant to scale changes, which was not the case for the structured scene type. The reason is that the textured wall$_1$ scene is captured fronto-parallel and for the structured door scene the camera is rotated of approximately 50 degrees

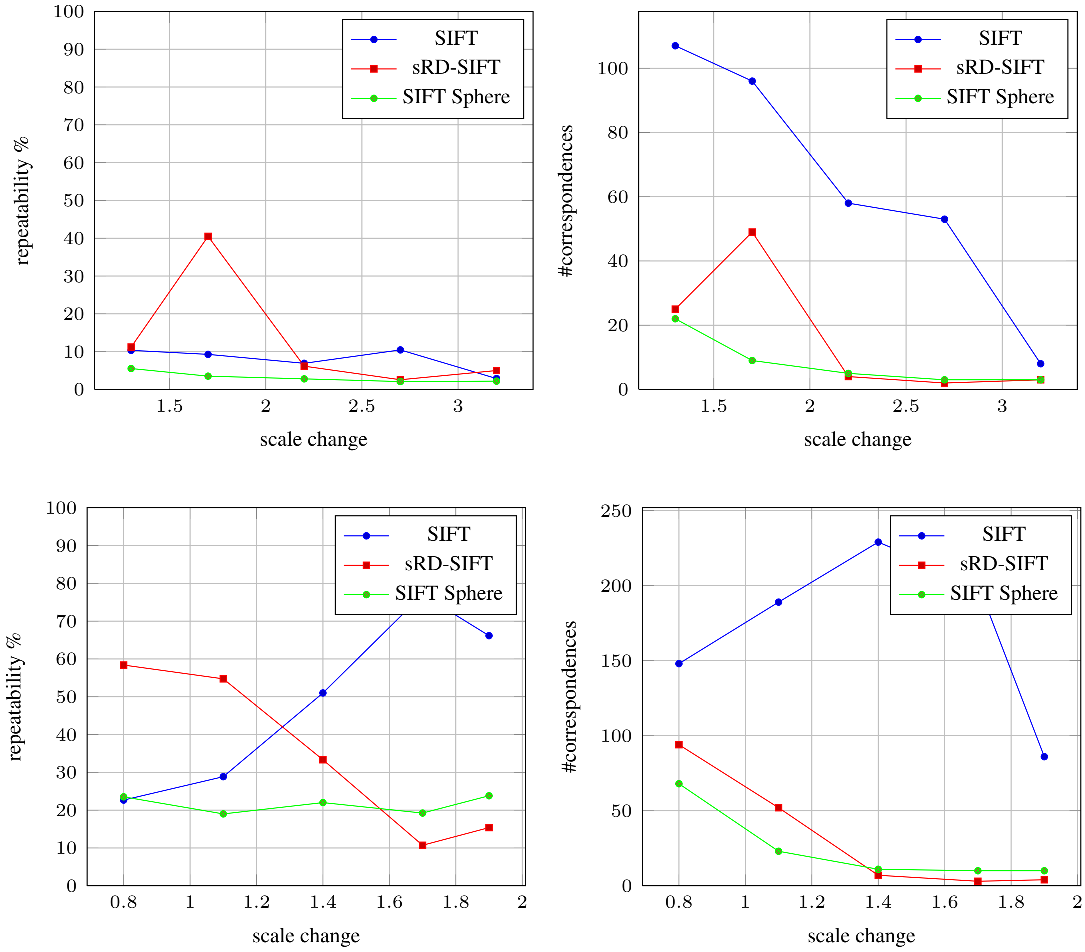


Figure 5.3: Scale changes for sparse textured scene (Wall$_1$ sequence Figure 4.1(b)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

against the plane. In the first case keypoints are predominantly detected in the center of the omnidirectional images and in the latter, the keypoints are lying closer to the image border and are therefore more affected by the non-linear distortion of the lens. The absolute number of corresponding keypoints decreases from 107 to 8 for SIFT and for sRD-SIFT and SIFT Sphere from 22, respective 25 to 3.

The effect of scale changes for the textured scene type in perspective to omni matching is different for each approach. For small scale changes of 0.8% the repeatability score of sRD-SIFT is 58%, which decreases for larger scale changes to 11%. Contrary to this, small scale changes in the case of SIFT result a relatively low repeatability score of 23% and then increases for larger scale changes to 77%. Apart from these numbers, SIFT Sphere is the only one which can keep invariance to scale changes, but with the exchange to a low repeatability score of 19% to 24%

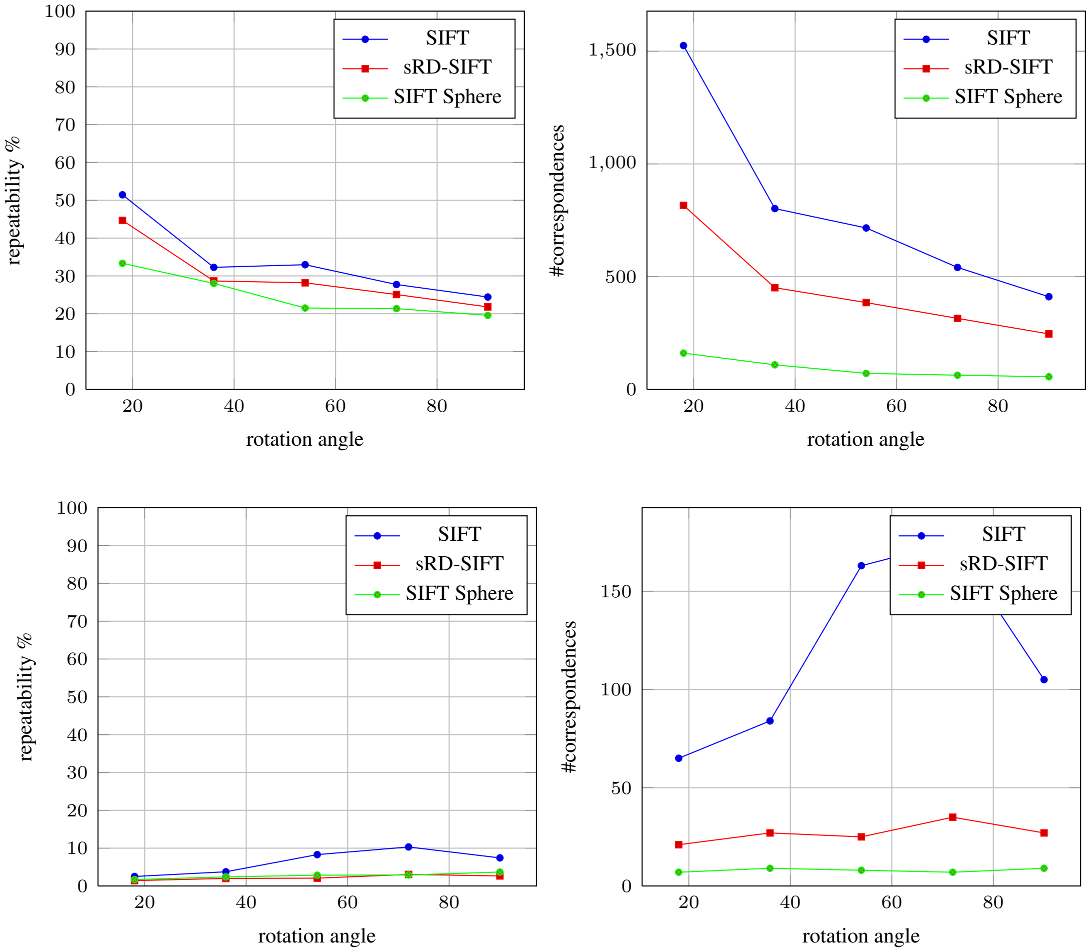


Figure 5.4: Field of view changes for sparse structured scene (Floor sequence Figure 4.1(c)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

for all scale changes. The number of detected correspondences ranges between 229 and 86. For small scale changes sRD-SIFT and SIFT Sphere start with 68 respective 94 correspondences. For large scale changes (factor 1.9) the detection shrinks down to 3 or 10 keypoints.

Field of View Changes

Figure 5.4 shows the results of the structured scene floor from Figure 4.1(c) for omni to omni matching and perspective to omni matching. In the first case the repeatability score has an approximately constant difference between SIFT, sRD-SIFT and SIFT Sphere. All are affected by field of view changes and the actual values are between 33% and 51% for a field of view change of 18 degrees. For an angle of 90 degrees they are between 20% and 24%.

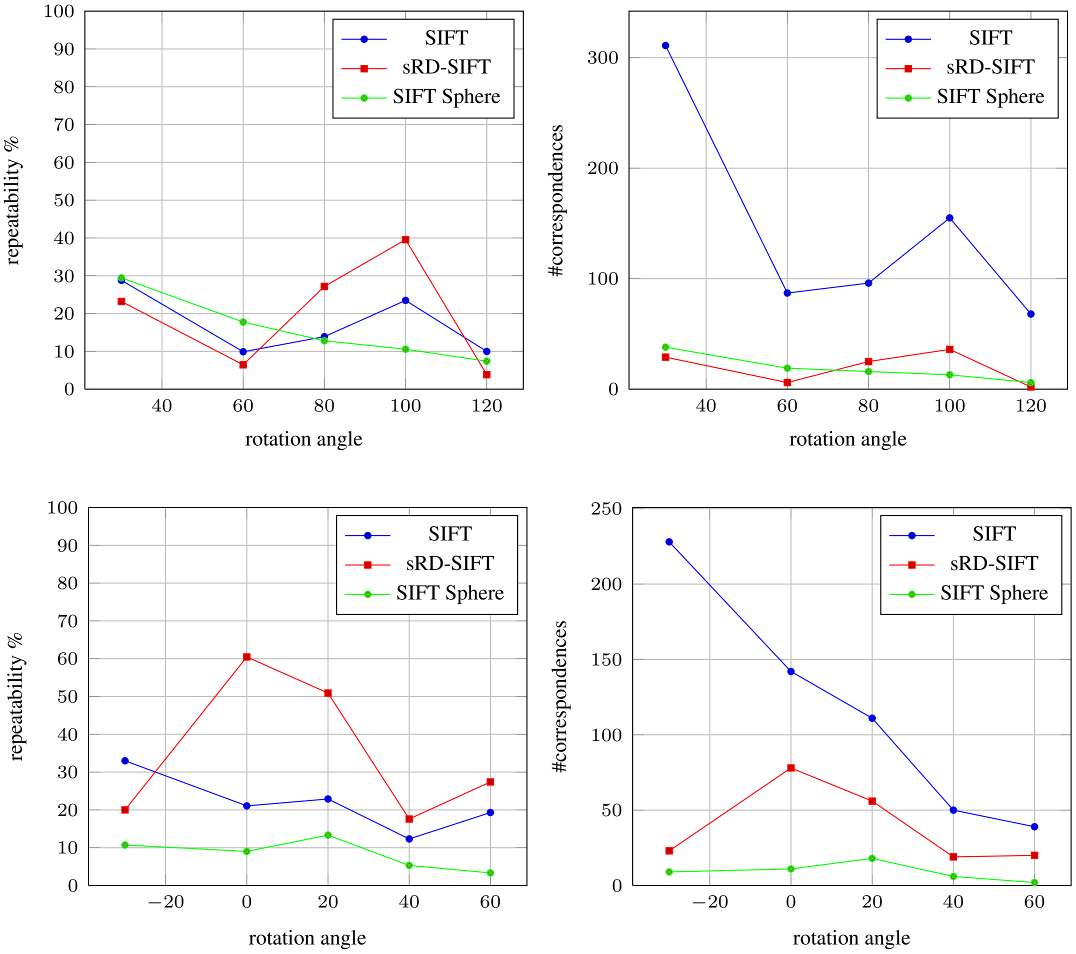


Figure 5.5: Field of view changes for sparse textured scene (Wall$_2$ sequence Figure 4.1(d)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

The results of SIFT are superior than the results of sRD-SIFT and again sRD-SIFT is on top of SIFT Sphere. However, SIFT Sphere is the least affected by the field of view changes. The floor sequence includes sparse, but very small structures. As a result most keypoints are extracted with a scale of 1 to 4 pixels. These points are minimally influenced by the radial distortion, since the spherical geometry locally reacts approximately like a planar surface (see Section 2.1). The number of detected correspondences results in the same ranking as for the repeatability score.

SIFT recognizes 1525 keypoints for small field of view changes and this number decreases to 161 for up to 90 degrees of field of view changes. Similar behavior is observed for sRD-SIFT within a range of 802 to 411 correspondences and for SIFT Sphere 161 to 56 correspondences. Here again, SIFT Sphere redetects at least keypoints because the original image is interpolated onto the sphere in a lower resolution. Especially in the floor sequence most of the structure is eliminated, since it corresponds to small scale values.

Perspective to omni matching again obtains a contrary result in comparison to omni to omni matching. sRD-SIFT and SIFT have only a maximal repeatability score of 3%. Similarly, the corresponding absolute number of redetected keypoints is low with a maximum of 35. The low results are due to the smaller overlap of the perspective image with the omnidirectional images rather than the overlap of two omnidirectional images with the same field of view change. SIFT performance is only slightly better, with a repeatability score up to 10% and 175 correspondences for a field of view change of 72 degrees. Again SIFT superior redetects the sparse and small image structure of the floor sequence better than the other approaches.

Floor scene	I_{ref}, I_2	I_{ref}, I_3	I_{ref}, I_4	I_{ref}, I_5	I_{ref}, I_6
omni to omni	18	36	54	72	90
perspective to omni	18	36	54	72	90
Wall$_2$ scene					
omni to omni	30	60	80	100	120
perspective to omni	-30	0	20	40	60

Table 5.2: Angles of field of view changes between images of the floor scene and wall$_2$ scene.

The effect of field of view changes for the textured wall$_2$ scene from Figure 4.1(d) is depicted in Figure 5.5. In the case of omni to omni matching the best results from 80 degrees to 120 degrees field of view changes are obtained by sRD-SIFT. Here the overlapping regions are lying in opposite image parts and both are highly affected by the radial distortion.

Since SIFT does not take the spherical distortion into account and SIFT Sphere loses the small image texture during interpolation, sRD-SIT is the only approach which can redetect such points. For small rotations between 30 degrees and 80 degrees, SIFT Sphere has the best repeatability rate. In contrast, sRD-SIFT provides the worst results with only 6% for changes of 60 degrees. The results obtained of SIFT lie between SIFT Sphere and sRD-SIFT.

Most interestingly all approaches are not totally invariant to field of view changes with the sparse textured scene type, but distinct ranking is observed. sRD-SIFT is mostly affected, followed by SIFT and the least affected is once again SIFT Sphere. The spherical model of the scale space is in this context again SIFT Sphere's strength. Despite the sparse texture, which includes straight lines and orthogonal structure only, SIFT can still detect 311 correspondences for 30 degrees field of view changes and 68 points for changes of 120 degrees. In comparison sRD-SIFT and SIFT Sphere can only redetect 38 to 2 keypoints.

Field of view changes regarding perspective to omni matching, gives a similar but more distinct observation. The mostly affected approach is once again sRD-SIFT and its results are in the most cases superior by comparison. The repeatability score is up to 60 degrees for non field of view change. Also without a field of view change the repeatability is not perfect, since the geometrical differences between the perspective image and the omnidirectional image is involved. SIFT Sphere is again the least affected by the field of view change, but also obtains a low repeatability score between 13% and 3%. The absolute number of correspondences is for all approaches for a field of view change of -30 degrees between 228 and 9 and for the maximum change of 60 degrees between 9 and 2. The ranking is the same as for the omni to omni matching.

Viewpoint Changes

Viewpoint changes for the structured scene type are evaluated with the entrance sequence from Figure 4.1(e). Of all sequences, this is the most difficult scene for matching. It contains just planar homogenous regions, with only two distinct borders and nearly no structure on the scene itself. During the estimation of the ground truth homography, only 11 corresponding points are found manually. Furthermore viewpoint changes correspond to the wide-baseline matching problem [49], since the distance between two viewpoints and their viewing angles increases drastically.

The results of keypoint detection for the structured scene related to viewpoint changes are shown in Figure 5.6. As expected the results are worse than in all other sequences, but still for this challenging task of sparse structure and wide-baseline matching, corresponding keypoints are found. Neither SIFT, sRD-SIFT nor SIFT Sphere are preferable in this case. The repeatability is between 13% and 17% for matching the second image of the sequence. This decreases to 2% and 3% for the image number five and finally slightly increases to 2% and 4% for the last image pair.

The numbers of corresponding keypoints begin with 29 to 13 keypoints, decrease to 2 and 5 keypoints, and finally increase again to 2 and 7 points for all approaches. The first decline can be explained by the increasing viewing angle between the features detected. The unexpected ascent in terms of correspondences and repeatability is caused by the wide viewing angle of the omnidirectional images. In this case an image structure appears and is detected, initially lying close to the image border and is therefore highly affected by the radial distortion. For increasing viewpoint changes this structure moves from the image border in the direction of the image center and is then matchable again.

In the case of perspective to omni matching, the results are similar but less affected by viewpoint changes. In other words SIFT, sRD-SIFT and SIFT Sphere are more invariant to viewpoint changes, than in the case of omni to omni matching. Again the absolute number of correspondences detected is low, starting from 4 to 2 keypoints for slightly viewpoint changes and 2 to 9 keypoints for large viewpoint changes (see Figure 4.1(e) for actual viewpoint change). The corresponding repeatability for image number 2 is approximately 2% for all approaches and varies until image number 6 between 2% and 4%. In this difficult scene, all matching approaches remain detected stable keypoints, because of the wide field of view of the omnidirectional camera. The few correspondences are detected constantly over the range of matching approaches.

Figure 5.7 shows the performances of the keypoint detector from SIFT, sRD-SIT and SIFT Sphere of the textured wall$_3$ scene from Figure 4.1(f) on viewpoint changes. The results are better than in the case of viewpoint changes on the structured scene. Again no method can be distinguished to give better results than the others in case of omni to omni matching.

An obvious difference, compared to the results of the structured scene, is the dependency on the viewpoint changes. The repeatability is in the range of 24% and 34% for image number two and decreases continuously to approximate 1% to 4% for image number six. The absolute

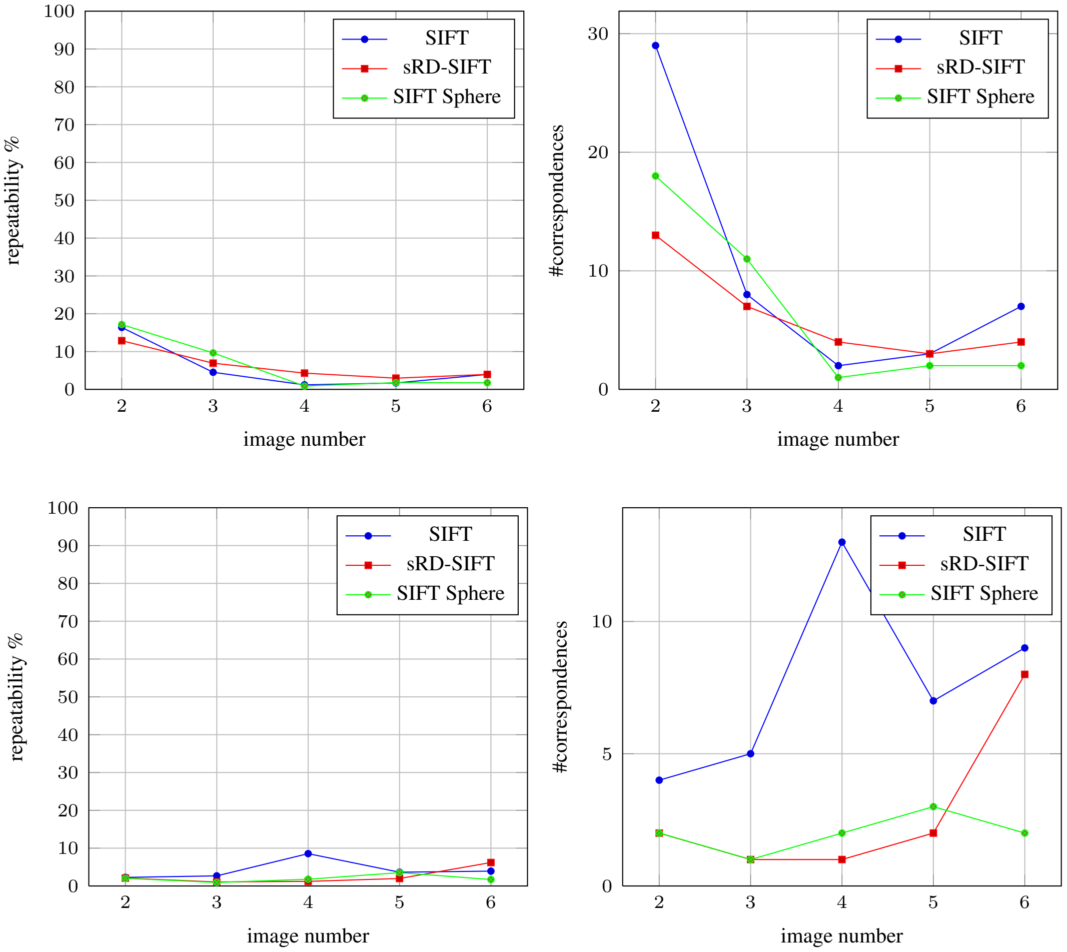


Figure 5.6: Viewpoint changes for sparse structured scene (Entrance sequence Figure 4.1(e)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

number of correspondences depicts a similar behavior, with the exception that SIFT starts with much more keypoints (264) than sRD-SIFT (38) and SIFT Sphere (13). The performance fits to the expectations of matching two perspective images with SIFT [29]. Omnidirectional vision does not affect viewpoint changes and the actual angle between two corresponding viewing rays still remains large for large viewpoint changes.

For perspective to omni matching similar results are obtained, i.e. showing a low level of invariance to viewpoint changes, except for the SIFT detector. Its repeatability is increased from image number two to image number four with 55% to 62%. Corresponding to the large number of redetected keypoints (303) this is clearly due to detection of keypoints with small scale values, which are approximately invariant to omnidirectional distortion. For sRD-SIFT and SIFT Sphere the repeatability starts between 22% and 28% and decreases to 0% and 8%.

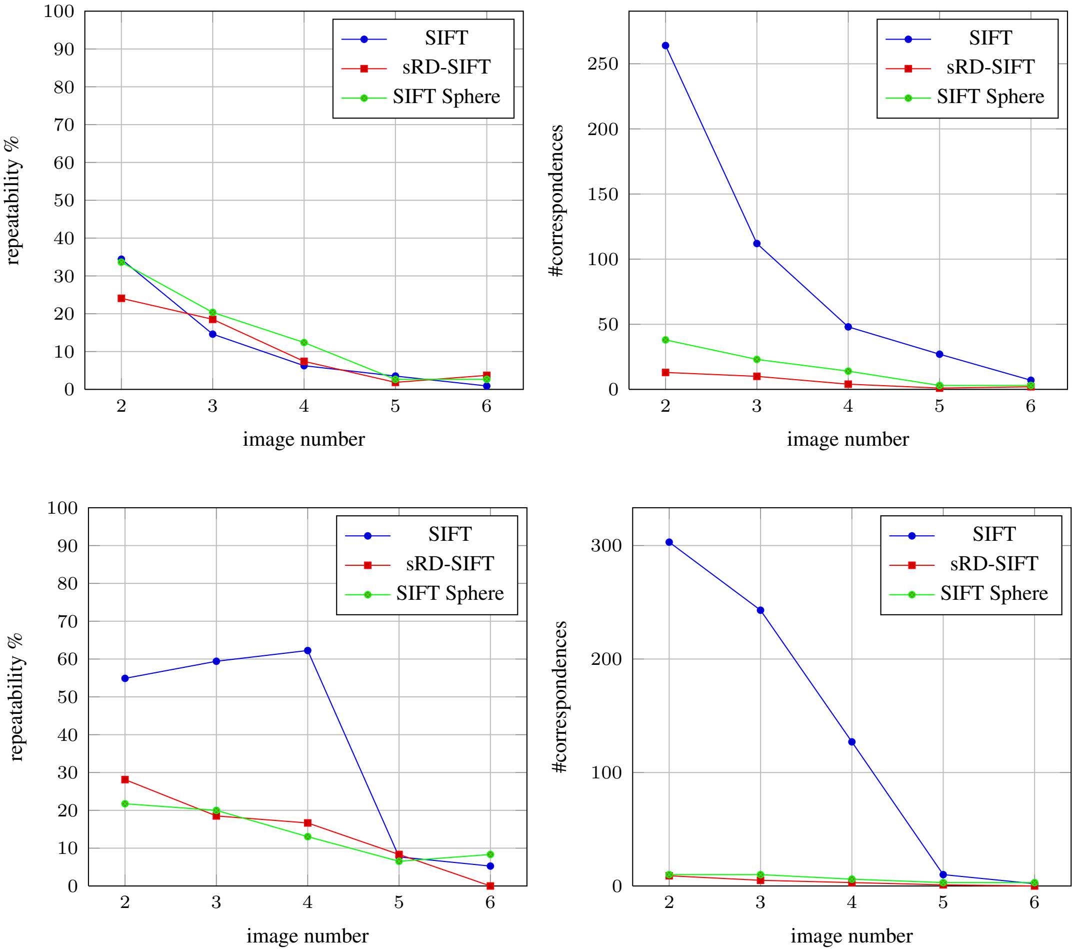


Figure 5.7: Viewpoint changes for sparse textured scene (Wall$_3$ sequence Figure 4.1(f)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

Thus, sRD-SIFT cannot redetect any keypoint for image number six. This is because the overlap of the two images remains only on the image boundaries. In that area the distortion model of sRD-SIFT is too inaccurate to detect any correspondence.

Rotation Changes

Figure 5.8 shows the results for rotation changes for the image scene ceiling from Figure 4.1(g). The ceiling sequence is the only one, rich in image structure and texture at the same time. Therefore only one scene is analyzed for this transformation.

The results between omni to omni matching and perspective to omni matching are very different in terms of detector dependency on the rotation angle. In the latter case, all approaches

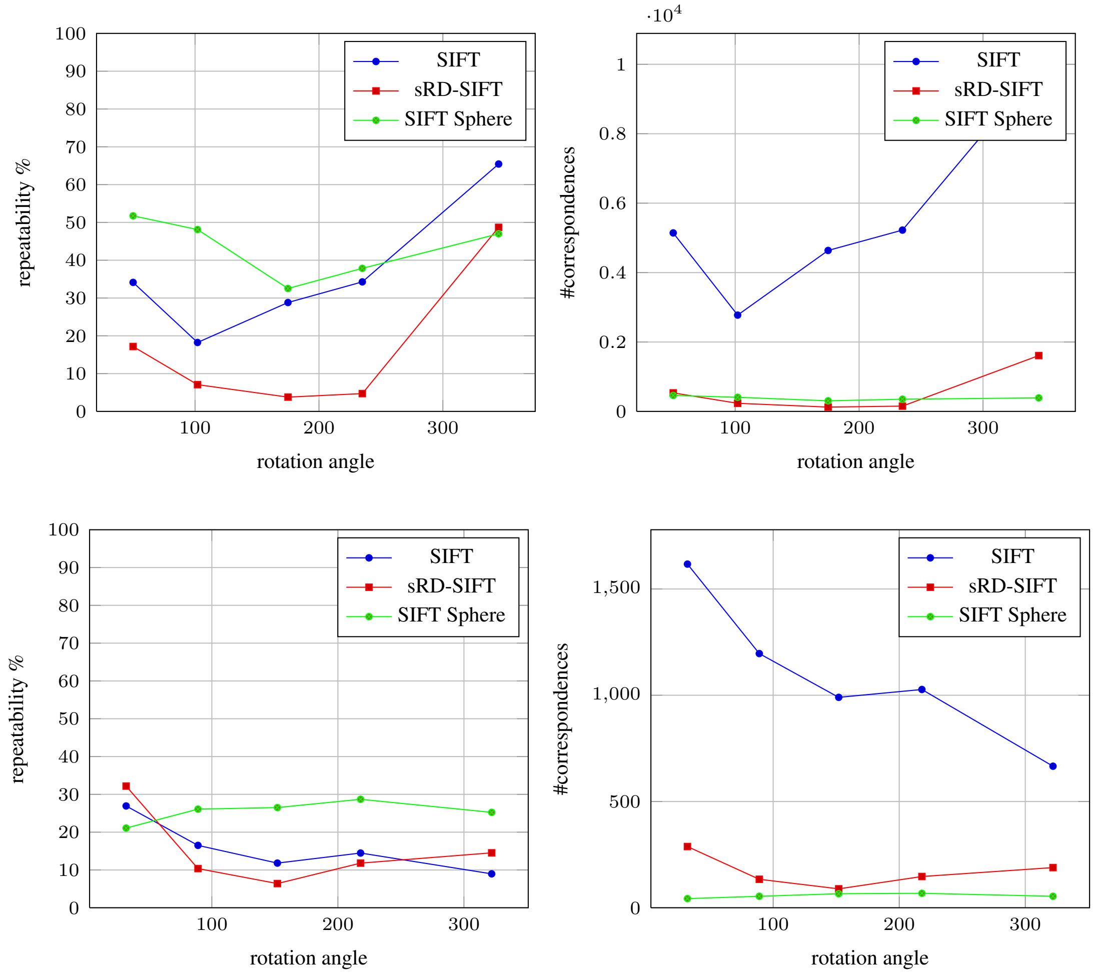


Figure 5.8: Rotation changes for textured scene (Ceiling sequence Figure 4.1(g)). (Top) Repeatability and number of corresponding points for omni to omni matching. (Bottom) Same for perspective to omni matching.

are more invariant to the transformation than in the first case. Due to the non linear image distortion of the omnidirectional image, images related to a rotation around the principle axis cannot be described by a linear transformation anymore. This is of course not valid for the perspective image and therefore the invariance can be kept better.

Overall in both cases SIFT Sphere obtains the best results, which are for omni to omni matching between 52% and 37% in repeatability. The number of correspondences is between 390 and 461 for all rotation angles. Because of this difference, even SIFT Sphere is in the case of omni to omni matching not completely invariant to rotation changes. The reason for this can be the insufficient detection accuracy of the keypoints, because of image resampling.

In contrast, sRD-SIFT is largely affected by the rotation and a small rotation angle (50 degrees) starts at 18% and decreases too only 4% for a rotation of 175 degrees. SIFT still can

Ceiling scene	I_{ref}, I_2	I_{ref}, I_3	I_{ref}, I_4	I_{ref}, I_5	I_{ref}, I_6
omni to omni	50	102	175	235	345
perspective to omni	32	89	152	218	322

Table 5.3: Angles of rotation changes between images of the ceiling scene.

redetect 29% of the keypoints for the same rotation angle. In the case of perspective to omni matching only SIFT and sRD-SIFT are affected by the rotation, and therefore SIFT Sphere provides the best results, which are still not more than 21% to 29%.

Discussion of Detector Performance

Finally it is concluded that of all approaches tested, neither is most superior compared to the others in indoor keypoint detection under the transformation considered. Additionally, there is not a unique best approach for each transformation type itself, e.g. the performance in the entrance sequence is equally for SIFT, sRD-SIFT and SIFT Sphere.

Overall, SIFT detects reliable keypoints in non linear distorted omnidirectional images, if the keypoints have a small scale value of about maximal 5 pixels or the keypoints are lying close to the image center. This is the case for the wall$_3$ sequence in perspective to omni matching. However, in general SIFT loses the invariances for transformations, scale changes and rotation changes. These are only feasible in linear perspective image matching. If invariance to those transformations can be achieved, then there is another transformation type to which a detector can be invariant for omnidirectional images. This is the field of view transformation, which corresponds to the rotation of the camera in any angle. Due to the underlying spherical model of omnidirectional vision, the image looks theoretically the same in every direction. However, in practice only the actual image resolution is not constant over the spherical surface.

As SIFT Sphere uses the spherical model, it provides more consistent results, compared to the other approaches over all transformations except for viewpoint changes. These cannot be compensated from the omnidirectional vision, because it only increases the field of view and does not affect the respective viewpoint. The drawback of SIFT Sphere is obviously the interpolation of the image into the spherical domain. This significantly affects the smaller scales and the interest points lying in those scales, can no longer be detected. Concerning indoor matching, the detection of keypoints with low scale value plays an important role as the sparse image structure does not provide rich content like the graffiti scene in [32], where much more keypoints can be detected. SIFT Sphere never detects more than 100 correspondences in each indoor sequence examined. In contrast SIFT detects, in some cases, more than 1500 points.

The performance of sRD-SIFT lies between both SIFT and SIFT Sphere. The weak interpolation is not included in this approach, but the underlying radial distortion model is only accurate enough in the central region of the image. In cases where only the image borders of two omnidirectional images are overlapping, e.g. in the wall$_3$ sequence due to viewpoint changes, sRD-SIFT cannot find any correspondence. For other transformations sRD-SIFT obtains the best results over all detectors e.g. the wall$_2$ sequence with field of view changes. Here the prominent image structure, where correspondences can be found, is localized away from the image borders.

Finding corresponding points in perspective and omnidirectional images compared to the omni to omni case is more difficult. In each image sequence the absolute number of keypoints and the repeatability score is worse, but still not zero. One exception is the wall$_1$ scene where the perspective to omni matching obtains superior results. Here the scale changes are in a range from a factor of 0.8 to a factor 2, whereby in the omni to omni matching case the range is from 1.5 to more than a factor of 3. Perspective images have a smaller field of view and what explains the lower performance in comparison to the omni to omni matching.

5.2 Results on Descriptor Matching

In this section the actual matching accomplished by each method is evaluated, regarding different image scenes and transformations. The measurement recall vs. 1-precision is used. An optimal curve would be a horizontal line, with recall equal to 1.0 for every precision between 0.0 and 1.0. If the orientation of the curve changes from horizontal to vertical, than the range of the resulting precision values become narrower. Note that for indoor matching, in multiple cases there are less than 30 keypoint correspondences, as shown in the previous section. This small number restricts the number of actual values of recall, and it results in a stair-like appearance of the recall precision curve. For example, possible recall values are 0.0, 0.5 and 1.0 for just two correspondences. The different data points of the curves are obtained by varying the threshold parameter of the matching strategy, which is examined in the next section.

The matching of keypoints is accomplished by matching their descriptors. Since the descriptors are estimated from the surrounding region of each keypoint, their matching performance depends on the keypoint detection performance. One method to only estimate the performance of the descriptor is to compare their performance on only one unique keypoint set. In the case of SIFT, sRD-SIFT and SIFT Sphere, this is not possible, since the descriptors are estimated in different image domains. In other words, each method is a specialized approach, where the descriptor estimation only works with the keypoints from their respective keypoint detectors. Therefore in the following descriptor evaluation, for each method their own keypoints are used. That means, the total matching performance of each matching approach is estimated and not only the sole performance of each descriptor. Of course, the matching performance still depends on the quality of the corresponding descriptor.

Actual matching results for the floor sequence are shown in Figure 5.9. The average scale of correct matched keypoints increases from SIFT, over sRD-SIFT to SIFT Sphere. Similarly, the number of correct matches decreases in the same order. Correct matches from SIFT and sRD-SIFT are concentrated in the image center, whereby the matches from SIFT Sphere are more equally distributed. The reason is, that for keypoints with larger scales the descriptor is not accurate enough, because of radial distortion. Superior matching performance is accomplished with the descriptors of sRD-SIFT. Only keypoints close to the image border remain un-matchable with sRD-SIFT. In contrast SIFT Sphere appropriates the spherical image model for calculating the descriptor and inaccuracies of the descriptors are appearing only from image interpolation. Therefore the descriptor matching for SIFT Sphere is not dependent on the image location of each keypoint. In general the number of matches is much lower for SIFT Sphere, than for SIFT or sRD-SIFT, since already the number of detected keypoints is less.

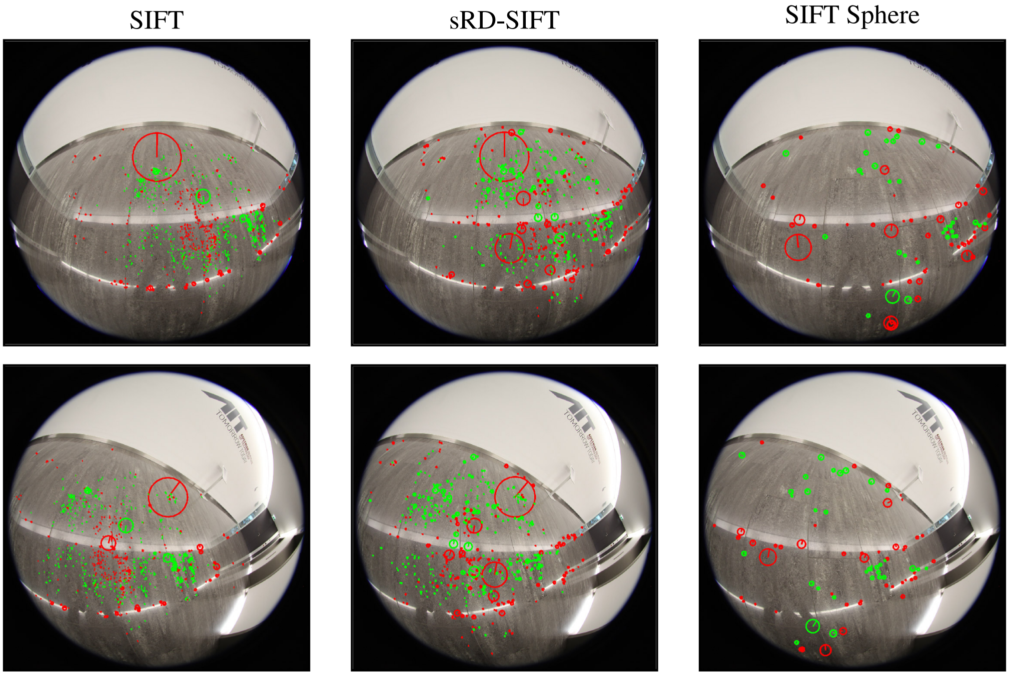


Figure 5.9: Keypoint matches from SIFT, sRD-SIFT and SIFT Sphere on the floor sequence. Incorrect matches are shown in red and correct matches are illustrated in green.

In the following subsections each image transformation is evaluated for keypoint detection and descriptor matching separately. Finally the estimated performance is discussed.

Matching Strategy

Before evaluating descriptor matching on different transformations, possible matching strategies are examined. As can be seen in [31] three of them are discussed on the basis of a specific application. In Figure 5.10 (top left) the matching results from threshold based matching on the door sequence are shown for SIFT, sRD-SIFT and SIFT Sphere. In this case there is a threshold which defines the maximal Euclidean distance for which two descriptors are still associated as a valid match. As a consequence, one descriptor can have several matches, but only one of them can be correct. Therefore the precision is low, but on the other hand a high recall up to 1.0 is reached. If multiple matches are allowed for one descriptor, then the probability that one of them are correct increases.

Another matching approach is nearest neighbor matching. Here the descriptor with the minimal Euclidean distance in descriptor space is associated as a match. Additionally, only matches under a certain threshold are taken. By changing the threshold, the curves in Figure 5.10 (top right) are established. The precision is slightly better than for threshold based matching, but the recall is lower. With this, fewer keypoints are incorrectly matched, but simultaneously not all correspondences are matched.

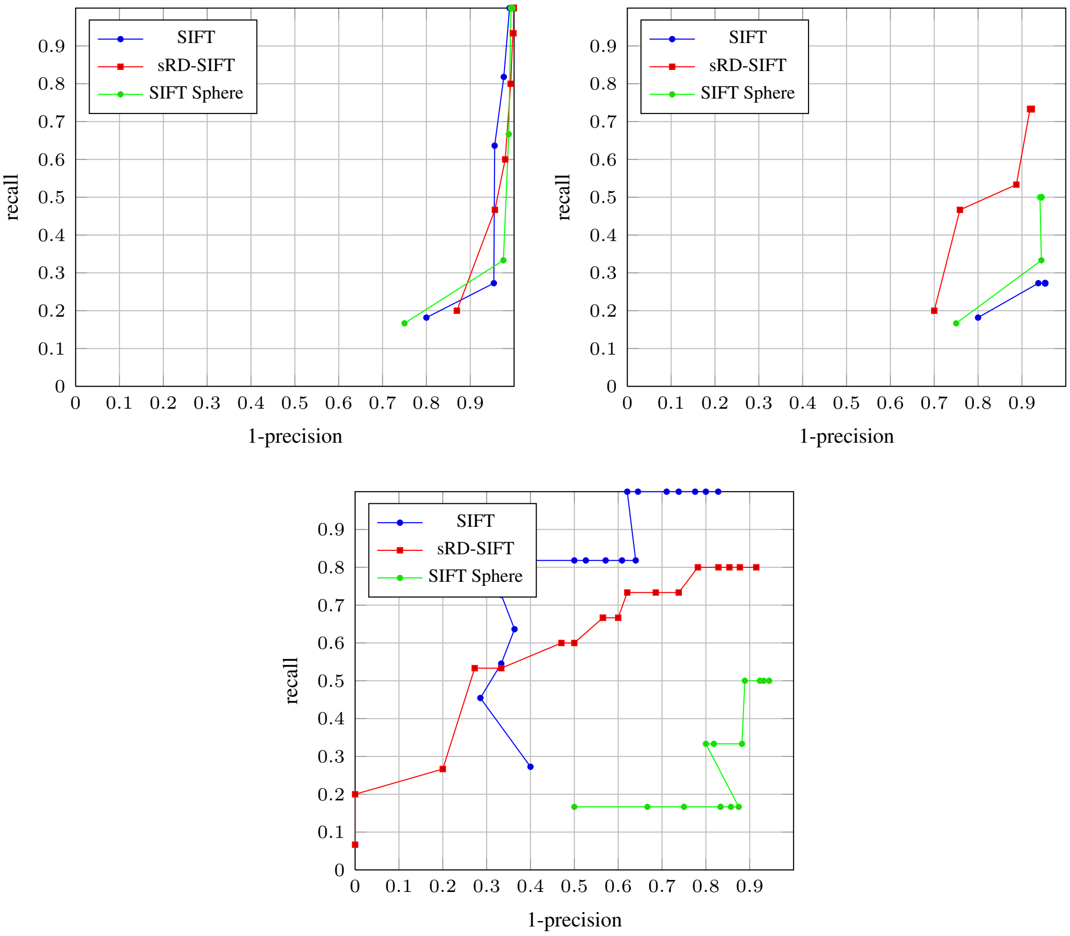


Figure 5.10: Comparison of matching performance of SIFT, sRD-SIFT and SIFT Sphere on image sequence Door (Figure 4.1 (a)) with descriptor matching strategies: threshold (top left), nearest neighbor (top right) and nearest neighbor distance ratio (bottom).

With the application of indoor matching the best matching results are achieved with nearest neighbor distance ratio matching. A descriptor is only matched if the ratio of the distance between the first and second best matches is smaller than a certain threshold. This strategy eliminates uncertain matches, where one descriptor has multiple, similar looking matching candidates. Specifically in indoor matching this is often the case, as images contain a lot of repeated structure and textured, e.g. tiles or panels. In Figure 5.10 the results from this strategy are shown in the plot at the bottom. A precision of 1.0 with a recall of 0.2 can be reached by sRD-SIFT for the specific image pair. For SIFT Sphere and sRD-SIFT a better precision is reached compared to the other matching strategies, i.e. 0.5 respective 0.3.

The ranking of the descriptors is similar for threshold-base and nearest neighbor distance ratio matching. Only for nearest neighbor based matching is the order different. As each descriptor is evaluated on its own keypoint set, the descriptor matching does not only depend on the descriptor itself. Where a different matching strategy is concerned, a particular matching approach can give better results than another with a separate matching strategy. Nevertheless, the results from nearest neighbor distance ratio based matching are suitably reliable to identify the best matching approach for a certain image pair, as the recall and precision is highly better than in nearest neighbor matching (Figure 5.10).

Scale Changes

In Figure 5.11 the descriptor matching performance of SIFT, sRD-SIFT and SIFT Sphere is shown for an image pair of the door scene and the wall$_1$ scene. Both pairs underlie a certain scale change which is 1.8 and 2.7 for omni to omni matching and 2.6 respective 1.7 for perspective to omni matching. The best matching results are achieved with the structured scene type (door scene). Here sRD-SIFT obtains the highest precision results, which are up to 1.0 and for lower precisions a recall of 0.8 in omni to omni matching and 0.4 in perspective to omni matching is obtained.

Better recalls are achieved using SIFT in both cases, but with the drawback of a worse precision, starting by 0.7 respective 0.5. Apart from that, the precision and recall values achieved by SIFT Sphere are much lower. With respect to omni to omni matching a recall of 0.5 is reached with a precision of only 0.1.

Perspective to omni matching is nearly impossible with SIFT Sphere on that scene, since the recall is only 0.1 with a precision of 0.1. The reason for these low results is the incorrect descriptors as shown in Figure 3.9. Especially in the case of perspective to omni matching, the LPD descriptor is computed on an interpolated region from a section with a larger scale value. Since regions with a larger scale value contain less information than regions with a lower scale value, a region cannot be correctly interpolated from one with a larger scale value. In case of omni to omni matching SIFT Sphere calculates the LSD descriptor directly on the spherical surface. Here the sole interpolation is carried out from the original image to the spherical surface, which results in better matchable descriptors than in the perspective case.

Matching of keypoints from the textured scene is a challenging task, since there are less than 20 correspondences for SIFT Sphere and sRD-SIFT. Only SIFT detects more than 200 correspondences in these scene. Surprisingly SIFT cannot match any of these correctly by the given matching strategy as shown in Figure 5.11. The reason is that each descriptor originates from a similar looking image patch. Since SIFT can only detect keypoints with small scale values properly (see Section 5.1), the descriptors of keypoints from the fine image structure of the wall$_1$ sequence are not distinguishable. SIFT Sphere is the only one which can detect reliable matches with larger scales in the spherical image domain. This approach is the only one, which can also match descriptors in the omni to omni matching case of the wall$_1$ scene. Accordingly the precision is low at approximately 0.8 with a related recall of 0.2. The best matching approach of the textured scene type for scale changes and perspective to omni matching is sRD-SIFT. Here, the precision obtained is between 0.5 and 0.1 and the corresponding recall starts from 0.1 and goes up to 0.6.

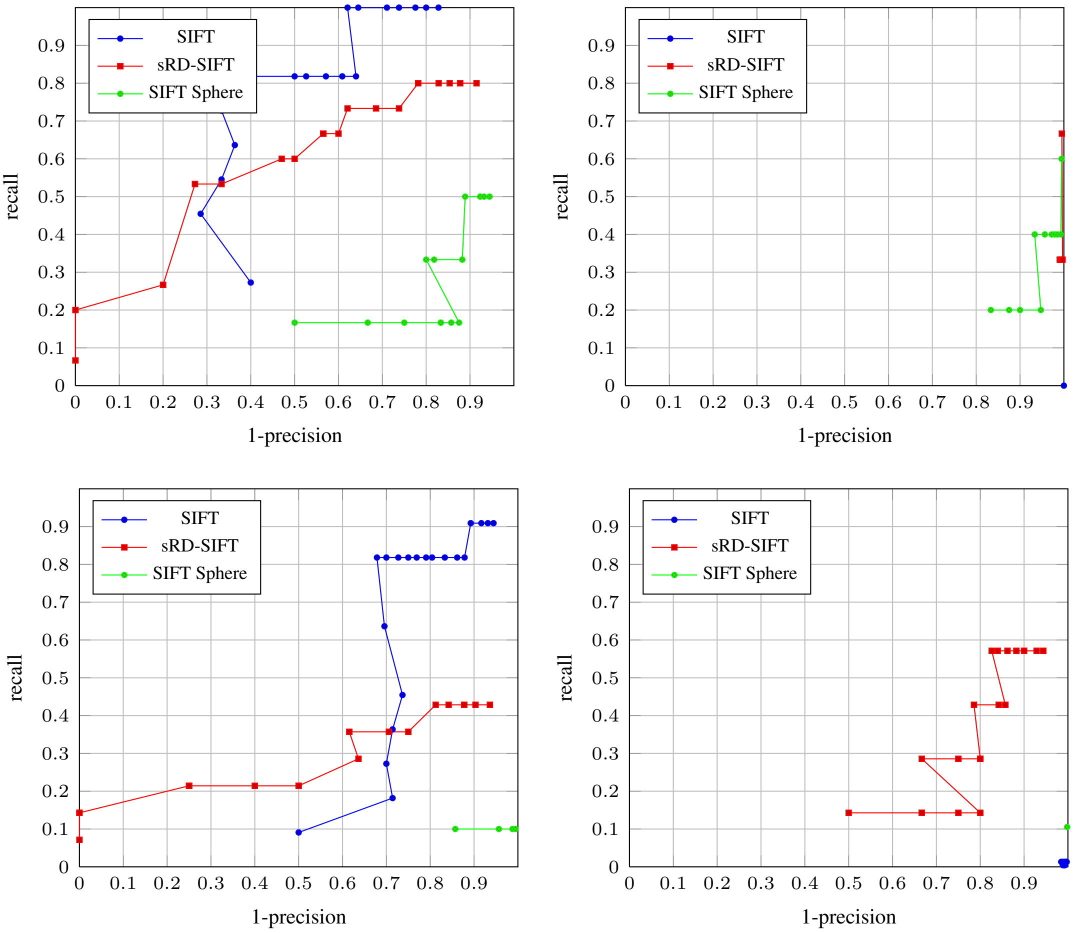


Figure 5.11: Scale changes of 1.8 degrees, respective 2.7 degrees for omni to omni matching (top) and scale changes of 2.6, respective 1.7 degrees for perspective to omni matching (bottom). Performance for structured scene (Door sequence Figure 4.1(a)) (left) and for textured scene (Wall$_1$ sequence Figure 4.1(b)) (right).

Field of View Changes

The results of descriptor matching on images with field of view changes between 40 degrees and 100 degrees are shown in Figure 5.12. The floor image sequence represents the structured scene type and the textured type corresponds to the wall$_2$ sequence. Both cases of omni to omni matching and perspective to omni matching are once again evaluated separately. The best matching results for the floor sequence are obtained by SIFT. This is because of the small distinctive structure available especially in the image center, where SIFT already achieved the best results on correspondence detection. The recall precision curve is relatively smooth in comparison to the other curves. The reason is the large number of correspondences detected in that scene by SIFT. In omni to omni matching this number is 716 and in the perspective to omni

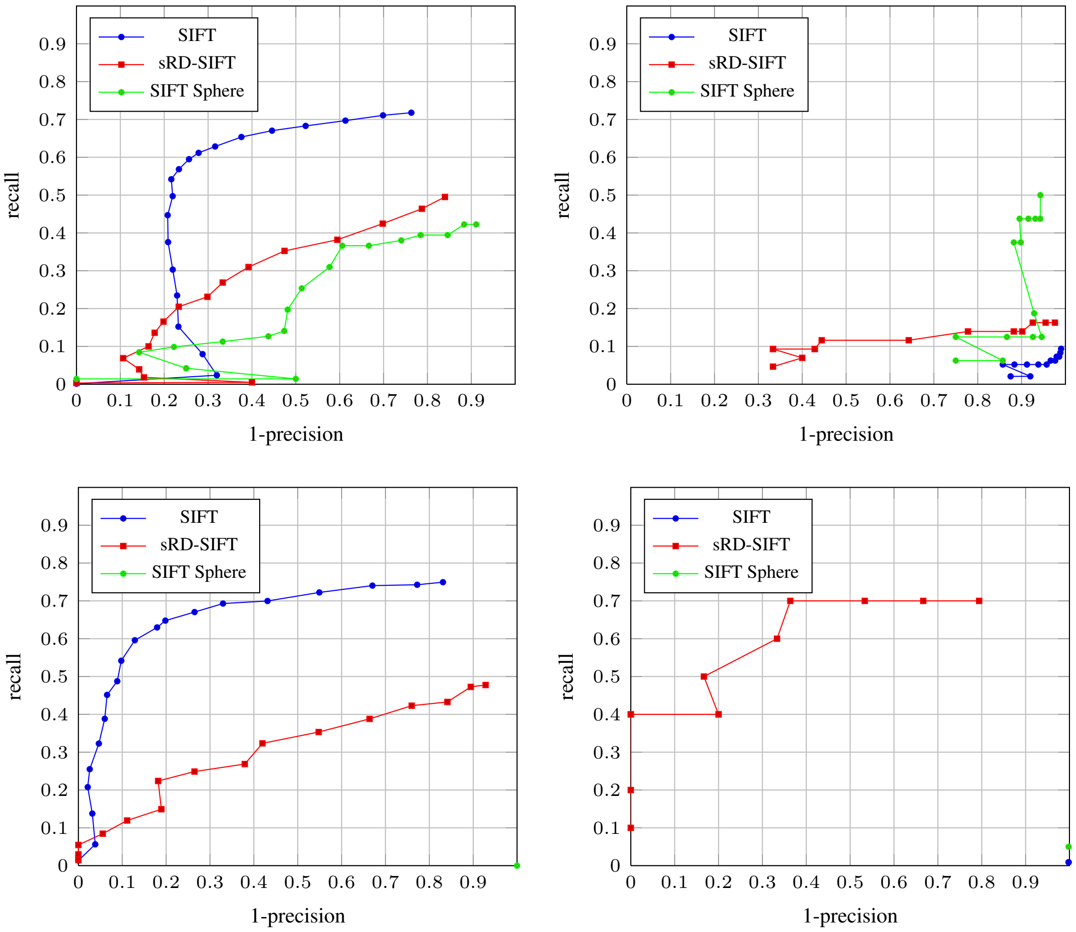


Figure 5.12: Field of view changes of 70 degrees, respective 100 degrees for omni to omni matching (left) and field of view changes of 70 degrees, respective 40 degrees for perspective to omni matching (right). Performance for structured scene (Floor sequence Figure 4.1(c)) (left) and for textured scene (Wall$_2$ sequence Figure 4.1(d)) (right).

matching it is still 175. The former results in a precision of 0.8 with a corresponding recall of 0.6 for SIFT. This is the best acquired matching result of all analyzed omni to omni matching scenes. An even better result is obtained by SIFT in the case of perspective to omni matching for the floor scene. Here the precision is even 0.9 for a recall of 0.6. sRD-SIFT obtains the second best matching results on the floor sequence which are still satisfactory. For a precision of 0.5 a recall of 0.4 is obtained in both matching cases. SIFT Sphere provides only moderate results in the omni to omni matching. In the perspective type no single match could be achieved. Again this is due to the poorly estimated planar descriptor.

Once again the textured scene type, i.e. the wall$_2$ sequence, is more challenging to match detected keypoints. In the perspective matching case only sRD-SIFT gives reasonable results, with a precision of 1.0 with a recall of 0.4. That means 40% of the 56 correspondences are correctly matched and no false positive match is included. For that scene SIFT do not find any correct matches and SIFT Sphere gives a precision of 0.01 with a recall of 0.05. These small values are negligible, as this corresponds to only one correct match.

In the case of omni to omni matching the results are much more scattered than in the previous case. Again, sRD-SIFT achieves the best result on precision, which is 0.4 for a recall of 0.1. In terms of recall, SIFT Sphere provides appropriate results. Here the recall is up to 0.5, but with a precision of less than 0.1. SIFT Sphere detects mainly larger regions which are in general more discriminative than smaller regions in the given indoor images. Therefore it is possible to actually match these corresponding keypoints. Nevertheless the precision is still low, because much more keypoints are matched incorrectly than correctly, due to the weak descriptor. In contrast the descriptor from sRD-SIFT is much more robust and detects less false positives from the wall$_2$ sequence, but yet cannot match much of the existing correspondence, since the recall is only 0.1. Here the reason is the repeated image structure which is still matchable especially with keypoints of small scales. These are also detected by sRD-SIFT. Accordingly SIFT gives the worst results for that image pair, i.e. precision equal to 0.01 with a recall of 0.1.

Viewpoint Changes

To evaluate the performance of viewpoint changes, images from the sparse structured entrance scene and from the sparse textured wall$_3$ scene are used. The entrance sequence is the most difficult case for finding corresponding points. For all image pairs, with the exception of the first, the number is below 11. In [31] it is observed that matching under viewpoint changes is already in the perspective case the most challenging transformation. Accordingly SIFT Sphere completely fails this task for the entrance scene, since it cannot find any correct match. After detecting only 4 correspondences, the descriptor is too inaccurate to be matched correctly.

In perspective to omni matching and omni to omni matching SIFT can match some of the few detections, with a precision of 0.2 and a corresponding recall of 0.5 in the former case, and with a constant recall of approximately 0.1 over precisions between 0.5 and 0.01. As sRD-SIFT do not give any correct matches also for the entrance scene for perspective to omni matching, SIFT is the best approach in this scenario. Again, the few existing keypoints comprise a small scale of only a few pixels. These are generally reliable, detected only by SIFT and can finally be matched. Because of the structured scene type, they are discriminative too. In the omni to omni matching case, keypoints also appear with larger scales. Once again, these are matched more favorably by sRD-SIFT than with SIFT, since the descriptor is more appropriate because of the underlying distortion model. Here the precision is low (0.1) but a relatively high recall of 0.5 is obtained. That means, most of the matched keypoints are false positives and are matched accidentally, because of similar image structure. Nevertheless, 3 of the 4 corresponding keypoints can be matched.

Also in case of the viewpoint transformation, the scene type has a high impact on the performance of a specific descriptor. With the exception of two previous transformations, scale and field of view changes, here the textured scene type gives better matching results than the structured type. This is because of the challenging matchable entrance scene, as shown above. Again SIFT Sphere only obtains correct matches in the omni to omni matching case. There the performance is, in terms of precision, in fact the best over SIFT and sRD-SIFT. It starts already

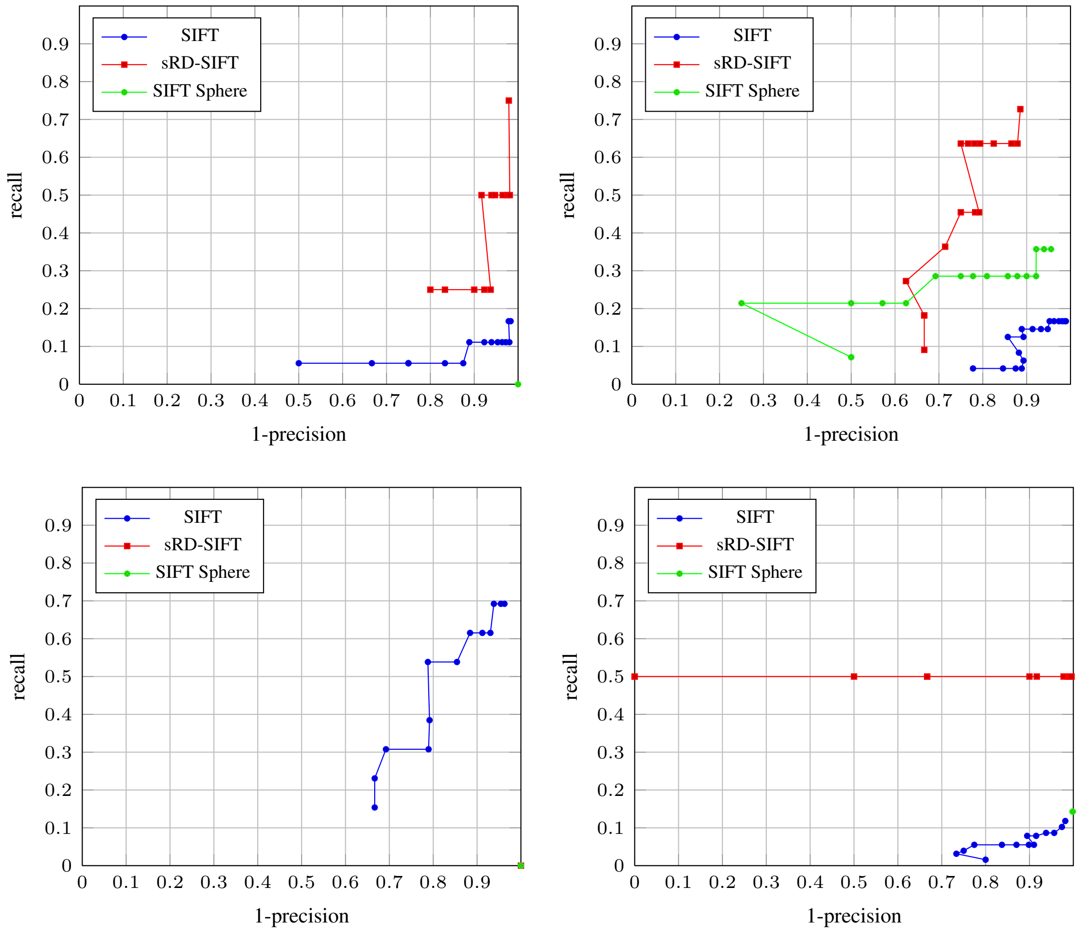


Figure 5.13: Viewpoint changes with the fourth image for omni to omni matching (top) and viewpoint changes with the fourth image for perspective to omni matching (bottom). Performance for structured scene (Entrance sequence Figure 4.1(e)) (left) and for textured scene (Wall$_3$ sequence Figure 4.1(f)) (right).

with 0.75 with a recall of 0.2. For lower precisions down to 0.1, the recall increases to approximately 0.4. It is concluded that only some of the existing corresponding keypoints relates to a uniquely descriptive image region, and therefore provide a more robust descriptor.

The descriptor of SIFT is due to radial distortion and repeated texture not accurate enough to provide correct matches for this image sequence. In comparison sRD-SIFT results in a better recall of up to 0.7 but in a worse precision of 0.3. That means, that the descriptor of sRD-SIFT is less influenced by the spherical distortion, since e.g. SIFT detects the most corresponding points but only gives a recall of 0.2 in descriptor matching. Additionally, SIFT does not even provide a better precision. The value of only 0.2 is again explained by the scene type. Similar results from SIFT are obtained for the same scene in perspective to omni matching. The precision is slightly better (0.3) but the recall is never larger than 0.1. The same problem as before plays a crucial

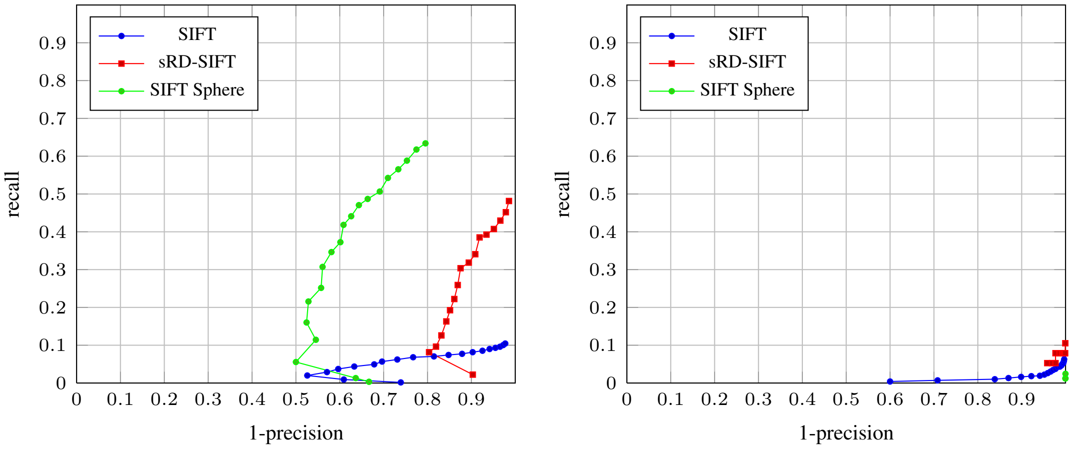


Figure 5.14: Rotation changes of 175 degrees for omni to omni matching (left) and for perspective to omni matching (right). Performance for textured scene (Ceiling sequence Figure 4.1(g)).

role in these results. SIFT Sphere matches maximum only one keypoint and sRD-SIFT gives an interesting result, because the recall is constantly 0.5 for all precisions from 0.0 to 1.0. The reason for this horizontal line is that there are only two corresponding points found, and only one is matchable.

Rotation Changes

Rotation change is the only transformation where all keypoint detectors are more or less affected. Therefore the keypoint matching demonstrates which matching approach has the best performance. The matching results of two images from the ceiling sequence under a rotation of 175 degrees are shown in Figure 5.14.

Concerning omni to omni matching SIFT Sphere and sRD-SIFT give rise to a similar recall precision curve, which has a larger variation in recall as in precision. For SIFT Sphere the precision is between 0.3 and 0.5 with a recall up to 0.6 and sRD-SIFT obtains a precision of 0.2 to 0.1 and a recall up to 5. Thus SIFT Sphere matches more keypoints than sRD-SIFT correctly. Therefore, only SIFT Sphere can handle rotation of 175 degrees with its spherical image model. Since the distortion model of sRD-SIFT is partially correct, the slightly but still good performance is reasonable. Apart from that, SIFT is lying in a similar precision range of 0.0 to 0.5, but obtains only a recall of 0.1 maximum. Most of the matches, especially keypoints detected in larger regions are not described accurately enough by the descriptor. In contrast sRD-SIFT and SIFT Sphere produces more robust descriptors.

Matching perspective and omnidirectional images under a rotation of 175 degrees is the most challenging task for the descriptors of the matching approaches analyzed (see Figure 5.14). It is the only case where none of the methods reach a better recall than 0.1 and a better precision than 0.4. Inside this small range SIFT can be identified as the best approach followed by sRD-SIFT and SIFT Sphere. It is concluded that the number of corresponding keypoints do not have to correlate with the number of actual matches, since SIFT detects more than 900 matchable points, and for sRD-SIFT it is still 89 and 66 for SIFT Sphere. In each case the worse matching results have a different reason. In the case of SIFT the descriptors from the omnidirectional image are incorrect, because of the ignored spherical distortion. sRD-SIFT is expected to detect more accurate descriptors, but in the case of the ceiling scene there are not only small but also larger very similar looking regions. Therefore most of the keypoints are accidentally wrongly matched. The same reason also holds for SIFT Sphere, but here the inaccurate local planar descriptor as discussed above is deciding.

Discussion of Descriptor Matching

Matching descriptors depends highly on the quality of the respective keypoint detector, since non detected keypoints cannot be matched. However, matching does not necessarily correlate with the performance of keypoint detection. The experimental evaluation shows there are cases where hundreds of correspondences are found, but only a few, e.g. 10, can be matched correctly by the descriptor. In contrast, the keypoint detector performs worse with only 10 correspondences for example, but then again the descriptor can still match most of them, e.g. 8. The result depends on the image transformation and the actual scene.

If the standard SIFT descriptor is used for matching omnidirectional images, it can provide a robust matching performance as shown for specific scenes. The best performance is obtained on the door scene and the floor scene. The images of these sequences are richer in fine image structure, e.g. pattern of the wooden door, or stone patterns of tiles, than the other scenes examined. SIFT highly depends on the scene type, because the results on the textured sequences are significantly worse. In some cases, particularly the wall$_1$ scene, less than 3 correct matches are established. In contrast, not only in the door and floor sequence, but also in the other structured scenes, SIFT provides the best matching results. In these cases most of the corresponding keypoints are still laying far from the image borders and the keypoint scales are small and only slightly affected by the spherical distortion. In cases where keypoints with larger scales have to be matched, typically the ceiling, SIFT fails due to the lack of a spherical distortion model.

In cases where the detected keypoints are lying on regions which are highly affected by the spherical distortion, particularly the image border or larger structures, then the best matching results obtained are with SIFT Sphere. For the sequences tested this is only the case in the ceiling and in the wall$_3$ scene. The major disadvantage of SIFT Sphere is the inaccurately interpolated keypoint descriptor, and therefore the results are worse compared to SIFT or sRD-SIFT. In particular, the local planar descriptor is based on a reasonable geometrical background and promises robust matching results. But in application of real world image matching it fails in all cases examined. For all transformations and all image types, there are never more than 3 correct matches found. In Figure 3.9 the influence of interpolation regarding the descriptor is shown.

sRD-SIFT plays an intermediate role between SIFT and SIFT Sphere. This is already established for its keypoint detector. It does not resample the image, and uses at the same time a distortion model. Unfortunately the distortion model is only partially correct. That means it applies well on the image center, but becomes inaccurate in direction to the image border. Also small keypoints are discarded because of non-computing of the octave -1. This is related to the default parameter set, and can be changed. sRD-SIFT is the sole approach which is only slightly affected by the scene type. The performance is between SIFT and SIFT Sphere, for example the door sequence, and it can keep the same performance in most of the textured scene types for omni to omni and for perspective to omni matching. There are only a few exceptions for which sRD-SIFT completely fails, i.e. does not match more than 1 keypoint correctly. This happens in the entrance scene for perspective to omni matching and in the wall$_1$ scene for omni to omni matching. Here the reason is a special keypoint constellation, where already only less than 4 correspondences are found with sRD-SIFT.

5.3 Overall Discussion

After comparing the performance of the state-of-the-art approaches SIFT, sRD-SIFT and SIFT Sphere on omnidirectional indoor image sequences, it is concluded that none of these methods is superior in general. Overall the performance is less favorable than in matching perspective images, as evaluated in [32]. On average a repeatability score of approximately 30% on structured scene, and a score of less than 20% on textured indoor scenes are achieved across all methods. This is relatively low in comparison to the 80%, proposed by Lowe [29]. The reason for such results is the dependency on the actual scene content and the actual accuracy of the spherical distortion model. Indoor scenery provides per definition less structure than natural outdoor scenes. But this fact means only that the absolute number of detected keypoints may be low, and not that the relative number of correspondences have to be necessarily low, too. Therefore a large potential for improvements on keypoint detection in omnidirectional images remains.

Room for improvement does not only exist for the keypoint detection, but also for the keypoint descriptors. The performance evaluation shows, that independent of the keypoint detector performance, keypoint matching can either be successful or not. The results highly depend on the actual scene type. With respect to the textured scenery including mainly repeated structure the performance is much lower than for images with non repeated structure. This problem appears by design already in perspective image matching, but in the case of omnidirectional matching in indoor environments it is even more challenging, since the structure diversity is lower and in the same time more of the same repeated elements are visible.

For the stability of the keypoint descriptor, it is observed that image interpolation, as done in SIFT Sphere, has a larger effect on performance, than non linear geometric deformations, which appears in the descriptor estimation for SIFT and partially for sRD-SIFT. Here the descriptor matching results are better. In particular these are images with fine non repeated structure, mainly located in the central region of the image. Keypoints with larger scales are less affected by the interpolation, but more influenced by the radial distortion. SIFT Sphere is the only approach which provides stable results concerning the matching of those keypoints. At the same time, these descriptors describe repeated structures better as they are, on larger scale, more diverse. It follows that SIFT Sphere should obtain the best descriptor matching results for the textured scene type, which is even reflected by the actual matching results.

It is shown that SIFT is not necessarily incapable of matching omnidirectional images. Without being designed for omnidirectional images, keypoints with a specific scale (<5 pixels) can be detected and matched by SIFT if the underlying image type allows a distinctive descriptor to be computed. Nevertheless SIFT loses most of its geometrical transformation invariances, i.e. scale and rotation changes. The influences from the transformations can only be removed if a proper model of the non linear image distortion is provided. SIFT Sphere recovers the initially proposed invariances on geometric transformations, but fails in detecting and matching keypoints from fine image structure.

All three approaches perform best using different ranges of keypoint scales. SIFT performs in small scales below 5 pixels, followed by sRD-SIFT which performs best with an intermediate range and SIFT Sphere achieve best matching results for large scale keypoints. This result depends not only on the detected keypoints, but also on the respective descriptor which only takes the spherical distortion in case of SIFT Sphere and sRD-SIFT into account.

Finally it depends on the specific application if choosing SIFT, sRD-SIFT or SIFT Sphere or even a combination of those three to obtain optimal matching results with omnidirectional images.

5.4 Summary

In this chapter, the evaluation of SIFT, sRD-SIFT and SIFT Sphere with omnidirectional images was given. Keypoint detection and descriptor matching was examined separately. Each approach had to perform under different image transformation, i.e. scale changes, field of view changes, viewpoint changes and rotation changes. Also a comparison between omnidirectional and perspective image matching was given for each scene type. For each transformation two scene types, i.e. structured and textured were analyzed. The keypoint detector performance was investigated with repeatability score in relative and absolute terms. For all transformations and scene types, also descriptor matching was evaluated with the measure recall vs. 1-precision. In this chapter it was shown, that none of the examined approaches gives the best results over all experiments. Additionally it is shown that only SIFT Sphere is invariant to the transformations examined.

CHAPTER 6

Conclusion

In this work three state-of-the-art approaches, namely SIFT, sRD-SIFT and SIFT-Sphere for matching omnidirectional images with the application of indoor environments were compared. An evaluation framework has been implemented, which enables the estimation of the matching performance of different keypoint detectors and descriptors on real world images. The main requirement of the framework is to identify if a point of interest detected in a reference image, is redetected on another image. Since SIFT keypoints contain not only a location, but a specific scale and orientation, those properties were also taken into account. With the identification of correspondences in two images, the actual descriptor matching performance is evaluated by comparing correct to incorrect matches. The accuracy of this evaluation approach depends on the accuracy of the underlying homography, which is used as a ground truth. An adoption of a state-of-the-art robust ground truth estimation approach for matching perspective images was proposed.

Previous work on matching perspective images have shown, that approaches for interest point detection and matching highly depend on the degree of image transformations, e.g. viewpoint changes, and scene types. Therefore, this work analyzed the given approaches under all Euclidean camera transformations, i.e. translation, scale. Their influences, not only on the keypoint detection but also on the final matching, were estimated for different scene types. The ability to estimate a robust ground truth was achieved by capturing planar scenes only. In contrast to perspective image matching, there is always more than only the planar surface visible in the wide field of view image. A special focus was given in this circumstance, i.e. different rotations and distances against the planar object were investigated.

Omnidirectional images yield special requirements on keypoint detection and descriptor estimations, since the images cover a field of view with more than 180 degrees, which can no longer be described by perspective geometry. Therefore two methods with contrary approaches, i.e. sRD-SIFT and SIFT Sphere, to handle the non-linear distortion, were selected to estimate the matching performance against their perspective equivalent, i.e. SIFT. A specific evaluation of both approaches was already separately provided by their corresponding authors in [28] and [8] respectively. In contrast, this work extended the evaluation environment to a larger image test set and to extensive performance estimation on the range of geometric transformations, i.e. translation, rotation, respective different scene types. This thereby improved the significance of the given results and provided more precise statements on conditional performance.

Omnidirectional cameras provide the opportunity to match images related to a much larger field of view change, contrary to perspective imaging. Secondly, more scene structure is observed and can lead to a more robust matching. From these properties omnidirectional images fit better than perspective images to match indoor scenery. In general, they contain only sparse structure e.g. walls, doors or floors. As a result, omnidirectional matching promises to achieve better matching results than perspective matching for the same application. The test data set for the evaluation used, included exclusively images captured indoor with sparse textured and structure. In case of the entrance scene even with omnidirectional images the approaches failed in matching this scenery, because no corresponding points were found by the approaches.

6.1 Overall Conclusion

The main insight of this work is that none of the approaches examined is preferable over the other. The overall performance remains significantly worse than that of detecting and matching interest points in perspective images. This comes from the fact, that none of the approaches examined entirely models the omnidirectional geometry and performs all image operations on the original raw image data. The matching performance of each method investigated highly depends on the image content and geometric transformation between the images matched. Each method investigated presents its own pros and cons and therefore provides the best matching result only in specific circumstances, e.g. in small structured scenery in case of SIFT.

The blind application of SIFT onto an image domain, for which is has not been designed, can still lead to promising results. The reasons are that keypoints with small spatial extent are, approximately, not influenced by the spherical distortion, since the sphere behaves locally like a plane. Secondly, the central region of the image is only slightly affected by radial distortion, and still with SIFT matchable. Nevertheless, in most cases where keypoints with larger scales all over the image are needed to be matched, SIFT is not capable in accomplishing that.

SIFT Sphere uses a spherical model to eliminate all of the radial distortion. The results are superior to SIFT, when keypoints with larger scale values need to be detected. Also the matches are equally distributed in spatial terms. The main disadvantage of SIFT Sphere is, that it fails to detect small scaled keypoints and to provide a proper descriptor. A reason for this is the interpolation of the original image data onto the sphere. Therefore SIFT Sphere redetects less keypoints then SIFT and sRD-SIFT, but it is the only approach which can keep approximately all geometric invariances proposed by SIFT and additional entire camera rotation invariance is obtained. Nevertheless, due to the poor descriptor quality the matching is significantly worse especially in matching perspective images with omnidirectional images.

sRD-SIFT does not interpolate the image data, but uses only an approximate model of the radial distortion. Therefore the descriptor estimated and keypoints are only partially accurate enough to obtain robust matching results. In cases where SIFT and SIFT Sphere fails, sRD-SIFT finds as the only one correspondences and correct matches, e.g. for the textured scenes.

It was shown, that matching cannot be improved under viewpoint changes with omnidirectional cameras. The larger field of view admittedly increases the overlapping region of two cameras from different viewpoints, but the viewing angle remains the same. The least favorable matching results, under large viewpoint changes of more than 50 degrees, are approved in case of omnidirectional matching.

6.2 Future work

Overall it is concluded that the methods examined are partially complementary and using multiple simultaneously should provide most robust results. Despite this, the performance of SIFT in perspective matching is not reached, and thus room for improvement prevails. An open question remaining is if this performance gap can be closed entirely with a merged approach of SIFT Sphere and sRD-SIFT, using a spherical model without resampling the image data.

If not, another question is if it is actually possible to obtain the same matching performance for omnidirectional images compared to that for perspective images.

In case of indoor matching, new approaches can be adapted to the sparse structure available in context of omnidirectional vision. With combined knowledge of the spherical distortion and the image elements, e.g. lines, homogenous regions, there could be an improvement in terms of finding stable image keypoints or regions.

Further work can focus on improving the matching performance of sRD-SIFT by using a more precise model of the non-linear distortion. An idea is to use the adapted inversion model from [30] for fisheye images, which depends on two parameters instead of one.

Bibliography

[1] Z. Arican and P. Frossard. Scale-invariant features and polar descriptors in omnidirectional imaging. *IEEE Transactions on Image Processing*, 21(5):2412–2423, 2012. 4

[2] S. Baker and S. K. Nayar. A theory of catadioptric image formation. In *Proceedings of the Sixth International Conference on Computer Vision*, pages 35–43, Washington, DC, USA, 1998. IEEE Computer Society. 11

[3] C. Belcher and Y. Du. Region-based SIFT approach to iris recognition. *Optics and Lasers in Engineering*, 47(1):139–147, 2009. 2

[4] A. J. Briggs, C. Detweiler, P. C. Mullen, and D. Scharstein. Scale-space features in 1d omnidirectional images. In *The Fifth Workshop on Omnidirectional Vision*, pages 115–126, 2004. 3

[5] M. Brown and D. Lowe. Invariant features from interest point groups. In *British Machine Vision Conference*, pages 656–665, 2002. 18

[6] T. Bülow. Multiscale image processing on the sphere. In *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, pages 609–617, London, UK, UK, 2002. Springer-Verlag. 3, 21, 22

[7] D. M. Chu and A. W. M. Smeulders. Color invariant surf in discriminative object tracking. In *Proceedings of the 11th European Conference on Trends and Topics in Computer Vision*, volume 2, pages 62–75, Berlin, Heidelberg, 2012. Springer-Verlag. 1

[8] 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*, 98(2):217–241, 2012. 1, 2, 3, 4, 10, 21, 22, 25, 26, 39, 40, 67

[9] K. Daniilidis, A. Makadia, and T. Bulow. Image processing in catadioptric planes: Spatiotemporal derivatives and optical flow computation. In *Proceedings of the Third Workshop on Omnidirectional Vision*, pages 3–11, Washington, DC, USA, 2002. IEEE Computer Society. 3, 10

[10] T. Dickscheid. *Robust Wide-Baseline Stereo Matching for Sparsely Textured Scenes*. PhD thesis, Institute of Photogrammetry, University of Bonn, 2010. 2, 3

[11] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 35

[12] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 125–132, 2001. 4, 14, 19

[13] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. Building rome on a cloudless day. In *Proceedings of the 11th European conference on Computer vision: Part IV*, pages 368–381. Springer-Verlag, 2010. 1

[14] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical implications. In *Proceedings of the 6th European Conference on Computer Vision*, pages 445–461, 2000. 11, 12

[15] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool. Omnidirectional vision based topological navigation. *International Journal of Computer Vision*, 74(3):219–236, 2007. 3

[16] E. B. Goldstein. *Sensation and Perception*. Wadswort Publishing, eighth edition, 2009. 1

[17] P. Hansen, P. Corke, and W. Boles. Wide-angle visual feature matching for outdoor localization. *The International Journal of Robotics Research*, 29(2-3):267–297, 2010. 1, 5

[18] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the Fourth Alvey Vision Conference*, pages 147–151, 1988. 35

[19] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 8, 32, 34, 35, 36

[20] E. Hecht and A. Zajac. *Optics*. Addison-Wesley, 1974. 11

[21] C. Hughes, M. Glavin, E. Jones, and P. Denny. Review of geometric distortion compensation in fish-eye cameras. In *IET Irish Signals and Systems Conference*, pages 162–167, 2008. 19

[22] Y. Kanazawa and K. Uemura. Wide baseline matching using triplet vector descriptor. In *Proceedings of the British Machine Vision Conference*, pages 28–38. BMVA Press, 2006. 1

[23] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50(5):363–370, 1984. 3

[24] S. Lazebnik, C. Schmid, and J. Ponce. A Sparse Texture Representation Using Affine-Invariant Neighborhoods Regions. In *International Conference on Computer Vision & Pattern Recognition*, volume 2, pages 319–324, Madison, United States, 2003. IEEE Computer Society. 1

[25] C.-F. Lee, S.-C. Wang, and Y.-J. Wang. Content-based image retrieval based on vector quantization and affine invariant region. In *Proceedings of the 2008 Eighth International Conference on Intelligent Systems Design and Applications*, volume 1, pages 287–290, Washington, DC, USA, 2008. IEEE Computer Society. 1

[26] L.-n. Li and N. Geng. Algorithm for sequence image automatic mosaic based on SIFT feature. In *Proceedings of the 2010 WASE International Conference on Information Engineering*, volume 1, pages 203–206, Washington, DC, USA, 2010. IEEE Computer Society. 2

[27] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 2(2):224–270, 1994. 3, 18

[28] M. Lourenço, J. P. Barreto, and F. Vasconcelos. sRD-SIFT: Keypoint detection and matching in images with radial distortion. *IEEE Transactions on Robotics*, 28(3):752–760, 2012. 1, 2, 3, 10, 16, 19, 20, 25, 37, 39, 40, 41, 67

[29] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2, 3, 10, 14, 15, 17, 18, 19, 23, 24, 37, 39, 40, 51, 65

[30] B. Micusik and T. Pajdla. Structure from motion with wide circular field of view cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1135–1149, 2006. 1, 33, 34, 69

[31] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005. 1, 2, 4, 23, 29, 32, 36, 38, 56, 61

[32] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005. 4, 29, 31, 32, 34, 35, 36, 43, 54, 65

[33] S. K. Nayar. Catadioptric omnidirectional camera. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, pages 482–489, Washington, DC, USA, 1997. IEEE Computer Society. 7, 8, 10, 11

[34] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, Washington, DC, USA, 2006. IEEE Computer Society. 1

[35] B. A. Olshausen and D. J. Field. What is the other 85% of V1 doing. In *in 23 Problems in Systems Neuroscience*, pages 182–211, 2004. 1

[36] D. D. Paola, D. Naso, A. Milella, G. Cicirelli, and A. Distante. Multi sensor surveillance of indoor environments by an autonomous mobile robot. *International Journal of Intelligent Systems Technologies and Applications*, 8(1–4):18–35, 2010. 2

[37] L. Puig and J. J. Guerrero. Scale space for central catadioptric systems: Towards a generic camera feature extractor. In *Proceedings of the 2011 International Conference on Computer Vision*, pages 1599–1606, Washington, DC, USA, 2011. IEEE Computer Society. 3

[38] B. Ruf, E. Kokiopoulou, and M. Detyniecki. Mobile museum guide based on fast SIFT recognition. In *Proceedings of the 6th international conference on Adaptive Multimedia Retrieval*, pages 170–183, Berlin, Heidelberg, 2010. Springer-Verlag. 2

[39] D. Scaramuzza, N. Criblez, A. Martinelli, and R. Siegwart. Robust feature extraction and matching for omnidirectional images. In *FSR*, pages 71–81, 2007. 1, 3

[40] C. Schmid, R. Mohr, and C. Bauckhage. Comparing and evaluating interest points. In *Proceedings of the Conference on Computer Vision*, pages 230–235, 1998. 2, 36, 37

[41] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, 2003. 1

[42] T. Svoboda, D. Martinec, and T. Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4):407–422, August 2005. 1

[43] J. D. Tardos, J. Neira, P. M. Newman, and J. J. Leonard. Robust mapping and localization in indoor environments using sonar data. *The International Journal of Robotics Research*, 21(4):311–330, 2002. 2

[44] F. F. Thomas Mauthner and H. Bischof. Region matching for omnidirectional images using virtual camera planes. In *Proceedings of the 11th Computer Vision Winter Workshop*, pages 93–98, 2006. 4

[45] A. Torii, M. Havlena, and T. Pajdla. From Google Street View to 3D city models. In *12th International Conference on Computer Vision Workshops*, pages 2188–2195. IEEE, 2009. 1

[46] T. Tuytelaars and K. Mikolajczyk. *Local Invariant Feature Detectors: A Survey*. Now Publishers Inc., 2008. 1

[47] A. P. Witkin. Scale-space filtering. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, volume 2, pages 1019–1022, San Francisco, CA, USA, 1983. Morgan Kaufmann Publishers Inc. 17

[48] L. Wolf and A. Zomet. Wide baseline matching between unsynchronized video sequences. *International Journal of Computer Vision*, 68(1):43–52, 2006. 1

[49] J. Xiao and M. Shah. Two-frame wide baseline matching. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, pages 603–609, 2003. 50

[50] Y. Yagi. Omnidirectional sensing and its applications. *IEICE Transactions on Information and Systems*, E82-D(3):568–579, 1999. 4, 10, 11

[51] X. Ying and Z. Hu. Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model. In *Proceedings of the 8th European Conference on Computer Vision*, pages 442–455. Springer, 2004. 12

[52] H. Zhou, Y. Yuan, and C. Shi. Object tracking using SIFT features and mean shift. *Computer Vision and Image Understanding*, 113(3):345–352, 2009. 2