



This is a repository copy of *FSD-BRIEF: a distorted BRIEF descriptor for fisheye image based on spherical perspective model*.

White Rose Research Online URL for this paper:  
<https://eprints.whiterose.ac.uk/173763/>

Version: Published Version

---

**Article:**

Zhang, Y., Song, J., Ding, Y. et al. (2 more authors) (2021) FSD-BRIEF: a distorted BRIEF descriptor for fisheye image based on spherical perspective model. *Sensors*, 21 (5). 1839. ISSN 1424-8220

<https://doi.org/10.3390/s21051839>

---

**Reuse**

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:  
<https://creativecommons.org/licenses/>

**Takedown**





If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing [eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk) including the URL of the record and the reason for the withdrawal request.



[eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk)  
<https://eprints.whiterose.ac.uk/>

## Article

# FSD-BRIEF: A Distorted BRIEF Descriptor for Fisheye Image Based on Spherical Perspective Model

Yutong Zhang <sup>1</sup>, Jianmei Song <sup>1</sup>, Yan Ding <sup>1,\*</sup>, Yating Yuan <sup>2</sup> and Hua-Liang Wei <sup>3</sup>

- <sup>1</sup> Key Laboratory of Dynamics and Control of Flight Vehicle, Ministry of Education, School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China; 3120160041@bit.edu.cn (Y.Z.); sjm318@bit.edu.cn (J.S.)
- <sup>2</sup> The Department of Applied Mathematics, The University of Waterloo, Waterloo, ON N2L 3G1, Canada; yating.yuan@uwaterloo.ca
- <sup>3</sup> Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 3JD, UK; w.hualiang@sheffield.ac.uk
- \* Correspondence: dingyan@bit.edu.cn

**Abstract:** Fisheye images with a far larger Field of View (FOV) have severe radial distortion, with the result that the associated image feature matching process cannot achieve the best performance if the traditional feature descriptors are used. To address this challenge, this paper reports a novel distorted Binary Robust Independent Elementary Feature (BRIEF) descriptor for fisheye images based on a spherical perspective model. Firstly, the 3D gray centroid of feature points is designed, and the position and direction of the feature points on the spherical image are described by a constructed feature point attitude matrix. Then, based on the attitude matrix of feature points, the coordinate mapping relationship between the BRIEF descriptor template and the fisheye image is established to realize the computation associated with the distorted BRIEF descriptor. Four experiments are provided to test and verify the invariance and matching performance of the proposed descriptor for a fisheye image. The experimental results show that the proposed descriptor works well for distortion invariance and can significantly improve the matching performance in fisheye images.

**Keywords:** fisheye camera; spherical perspective model; distorted BRIEF descriptor; feature point attitude matrix



**Citation:** Zhang, Y.; Song, J.; Ding, Y.; Yuan, Y.; Wei, H.-L. FSD-BRIEF: A Distorted BRIEF Descriptor for Fisheye Image Based on Spherical Perspective Model. *Sensors* **2021**, *21*, 1839. <https://dx.doi.org/10.3390/s21051839>

Academic Editor: Omar Ait Aider

Received: 9 January 2021  
Accepted: 25 February 2021  
Published: 6 March 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

For decades, feature detection and matching is one of the core areas of image processing in various applied fields, such as Visual based Simultaneously Localization and Mapping (V-SLAM), Structure from Motion (SfM), Augmented Reality (AR), general image retrieval, image mosaic, and image registration. Common features include Scale Invariant Feature Transform (SIFT) [1], Speed Up Robust Feature (SURF) [2], BRIEF [3] Oriented FAST and Rotated BRIEF (ORB) [4], KAZE [5], Binary Robust Invariant Scalable Keypoints (BRISK) [6], etc. and their derivations, such as Principle Component Analysis SIFT (PCA-SIFT) [7], Simplified-SIFT (SSIFT) [8], and Accelerated-KAZE (AKAZE) [9]. Neural network based features are also developed, such as L2-NET [10], HardNet [11], and AffNet [12]. These features are designed for pinhole images with little distortion and cannot achieve good performances for fisheye images with severe radial distortion.

Compared with a pinhole camera, a fisheye camera has a wide field of view (FoV), and the captured image contains more abundant information. This makes the fisheye camera extensively adopted in robot navigation, visual monitoring, virtual reality, visual measurement, and 3D reconstruction. However, due to the severe radial distortion of the fisheye image, adopting the common feature descriptors directly may lead to a significant reduction in matching performance.

In order to reduce the impact of distortion on the feature matching performance, we propose a novel distorted BRIEF descriptor based on the spherical perspective model,

named Fisheye Spherical Distorted BRIEF (FSD-BRIEF). Firstly, we propose a method based on 3D gray centroid to determine the direction of each feature point in the spherical image. By constructing an attitude matrix of a feature point, the position and direction of the feature point in the spherical image can be described in a nonsingular form. In order to reduce the calculation error of the 3D gray centroid caused by uneven distribution of pixels in the spherical image, a pixel density function is designed to represent the degree of pixel density on the spherical surface by the size of the patch area mapped by each pixel in the fisheye image. We build an attitude coordinate system of each feature point and propose a coordinate mapping method to project the BRIEF descriptor template on the fisheye image. The distortion form of the projected BRIEF template is consistent with the image distortion near the feature point, which prevents the calculated BRIEF descriptor from the affection of the radial distortion in fisheye image. The main contributions of the paper include:

1. A new pixel density function represented by the area of the spherical surface patch that each pixel of fisheye image occupies;
2. A new method of determining the 3D gray centroid and the direction of feature points with pixel density function based on a spherical perspective model;
3. A new feature point attitude matrix, providing a nonsingular description for both the position and the direction of the feature point in the spherical image surface;
4. A novel descriptor template distortion method based on the spherical perspective model and the feature point attitude matrix.

The remaining of the paper is arranged as follows. In Section 2, the related work of the fisheye image point feature is presented. In Section 3, the notation of the perspective model is briefly introduced. Section 4 is about the method of determining and expressing the direction of feature point. Then the method of calculating the FSD-BRIEF descriptor is described. In Section 5, experimental results are provided and the performance of the proposed FSD-BRIEF is tested and verified. Section 6 briefly summarizes the work. In Section 7, the future work is stated.

## 2. Related Work

By virtue of its front lens protruding in a parabola shape, fisheye camera has a large FoV whose angle of view is close to or even more than  $180^\circ$ . Although this characteristic can maximize the angle of view, it brings severe radial distortion in its captured image, leading to different scale factors for pixels in different positions of the image. Thus, it could make the traditional feature descriptors designed for plane image fail to match in raw fisheye images [13,14].

Generally, the methods to extract descriptors in fisheye images can be divided into two main streams according to whether images are corrected or not: resampling and non-resampling approaches.

Resampling approaches [15–17] segment the FoV image into several sub-FoVs and correct them based on a plane perspective model, then feature descriptors can be extracted and matched on the corrected sub-FoV. Lin et al. [15] adopted a visual-inertial based UAV (Unmanned Aerial Vehicle) navigation system, where two sub-regions are sampled in the horizontal direction of the fisheye FoV to obtain two undistorted pinhole image fields, which cover  $180^\circ$  horizontal FoV, but they discarded the upper and lower parts of the vertical FoV. Miiller et al. [16] presented a robust visual inertial odometry and time-efficient omni-directional 3D mapping system, where the FoV of each fisheye camera is divided into two piecewise pinhole fields so as to overcome the distortion. However, some parts near the edge of the FoV are wasted. Wang et al. [17] proposed a new real-time feature-based simultaneous localization and mapping system, where a fisheye image is projected onto five surfaces of a cube, and then descriptors are extracted on the unfolded surfaces of the cube. However, the stretching distortion and seam distortion exist between surfaces, for example, a straight line will become a broken line. Thus, in the resampling approaches, the whole FoV of the fisheye image is hard to be fully utilized, and the continuity between sub-FoV

cannot be guaranteed. In addition, due to the view geometry of the plane perspective model, there is a small stretching distortion in the edge of the sub-FoV.

Unlike the resampling approaches, which directly correct fisheye images to pinhole images, a non-resampling approach uses descriptors to describe features in fisheye images. For example, inspired by the planar SIFT framework [18–20], Arican et al. [21] designed a new scale invariant omni-directional SIFT feature based on Riemannian geometry. Lourenco et al. [22] proposed a Spherical Radial Distortion SIFT (sRD-SIFT) feature, where the extraction of the feature and the calculation of the descriptor was designed based on the spherical perspective model and the raw fisheye image without resampling. However, the improved algorithms based on SIFT are generally long time-consuming. Cruz-Mota et al. [23] and Hansen et al. [24] utilized spherical harmonic function as the basic function to study the spectral analysis of spherical panoramic images. Since Gaussian filtering on the sphere can be realized as a diffusion process through the spherical Fourier transform, spherical harmonic function is used to construct scale space on the sphere. In theory, the spherical harmonic function can be used to maintain the invariance of the descriptors to encounter the changes of the camera poses and positions. However, the spherical harmonic function usually needs a large amount of computation and has inherent bandwidth limitation. This greatly weakens the capability of dealing with large-scale matching problems and cannot meet the real-time requirements of many applications.

For improving the calculation speed, Qiang et al. [25] proposed Spherical ORB (SPHORB), a binary spherical feature based on the ORB feature, which is the first binary descriptor for a panoramic image based on hexagon geodesic grid. In essence, SPHORB is still a special resampling approach, which divides the spherical panoramic image into 20 regular triangle fields according to the shape of a regular icosahedron, and aligns the pixel of adjacent regular triangles seamlessly. However, in the hexagon geodesic grid, the image patches near the 12 vertices of the regular icosahedron are discarded due to the distortion of the pixel distribution pattern, resulting in 12 FoV holes occupying 1.4% of the total FoV.

Note that it can result in holes when resampling the fisheye image based on hexagon geodesic grid. To avoid this, Urban et al. [26] proposed a new distorted descriptor, called Masked Distorted BRIEF (mdBRIEF). Although this work distorts the descriptors to adapt to different image regions instead of correcting the distortion of the fisheye image, the direction angle of feature points is obtained in the raw fisheye image by calculating the gray centroid in a circle template, which is still affected by the fisheye image distortion. Furthermore, the descriptors are distorted excessively near the edge of the fisheye image since it is distorted based on the plane perspective model.

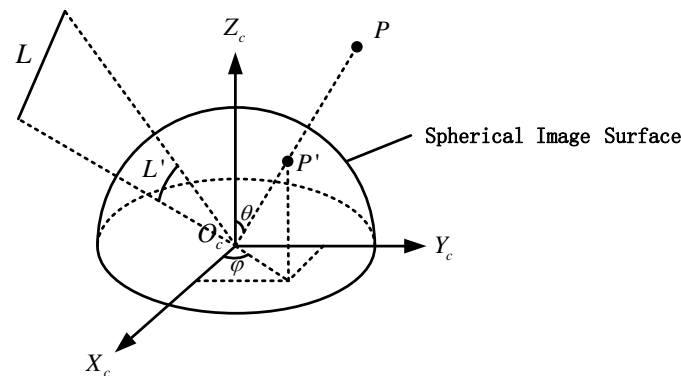
Most recently, Pourian et al. [27] proposed an end-to-end framework to enhance the precision of the descriptor matching between multiple wide-angle images. In their work, the global matching and the local matching of descriptors are combined in three stages. However, a new distortion in the edge of the corrected image is introduced when an equal rectangle image transformation is employed in the global matching stage, lowering the performance of the framework.

In summary, the binary descriptor that can make use of the whole FoV and keep invariance in each position of the fisheye image has not been proposed. In order to avoid the FoV holes caused by the resampling approaches, and reduce the excessive distortion of descriptors in large FoV images, in this paper, we design a novel Fisheye Spherical Distorted BRIEF (FSD-BRIEF) descriptor, which is a distorted binary feature descriptor based on the spherical perspective model for fisheye images.

### 3. Fisheye Camera Model

In this paper, in order to ensure that the FoV of the fisheye image can be fully utilized without losing the performance of the feature descriptor, a new descriptor FSD-BRIEF is designed based on spherical perspective model. Different from the plane perspective model, the projection surface is a unit sphere with the origin of camera coordinate system as center, so as to ensure that the scale factors of each position on the projection surface are

consistent. The spherical perspective model and its perspective projection relationship are shown in Figure 1. We define the camera coordinate system as  $O_c X_c Y_c Z_c$ . The origin point  $O_c$  is located at the optical center of the camera, the X-axis  $O_c X_c$  points to the right along the long side of the imaging target surface, the Y-axis  $O_c Y_c$  points downward along the wide edge direction of the imaging target surface, and the Z-axis  $O_c Z_c$  points to the front of the camera along the optical axis direction.  $P'$  is the projection point on the spherical image surface of the space point  $P$ .  $L'$  is the projection large arc on the spherical image surface of the space Line  $L$ .



**Figure 1.** Spherical perspective model ( $\theta$ : The FoV latitude angle;  $\varphi$ : the FoV longitude angle).

For a point  $P$  in a three-dimensional space, define its space coordinate in camera coordinate system as:

$$\mathbf{P}_c = [x \quad y \quad z]^T \quad (1)$$

The projection point of  $P$  in the fisheye image is  $\mathbf{p}$ , and its pixel coordinates are expressed as follows:

$$\mathbf{p} = [u \quad v]^T \quad (2)$$

In this paper, Kannala-Brandt4 (KB4) [28] model is used as the fisheye camera model, its mathematical form is shown below:

$$\begin{aligned} \theta &= \arctan 2(\sqrt{x^2 + y^2}, z) \\ \varphi &= \arctan 2(y, x) \\ \theta_d &= \theta(1 + k_1\theta^2 + k_2\theta^4 + k_3\theta^6 + k_4\theta^8) \\ u &= f_x\theta_d \cos \varphi + c_x \\ v &= f_y\theta_d \sin \varphi + c_y \end{aligned} \quad (3)$$

where  $f_x$  and  $f_y$  are the horizontal and vertical focal length of the camera,  $c_x$  and  $c_y$  are the coordinates of the principal points of the camera, and  $k_1, k_2, k_3, k_4$  are the distortion coefficients.  $\theta$  is the FoV latitude angle, which represents the angle between the  $O_c Z_c$  axis and the vector  $\overrightarrow{O_c P}$ .  $\varphi$  is the FoV longitude angle, which denotes the angle between the  $O_c X_c$  axis and the projection vector of  $\overrightarrow{O_c P}$  on the  $X_c O_c Y_c$  plane.  $\theta_d$  is the angle  $\theta$  as deflected by the fisheye lens. The  $\arctan 2$  is the quadrant aware version of arctangent function.

Based on the spherical perspective model in Equation (3),  $\Pi$  represents the mapping function. The mapping from the point  $\mathbf{P}_c$  to the pixel point  $\mathbf{p}$  in fisheye image can be expressed as:

$$\mathbf{p} = \Pi(\mathbf{P}_c) \quad (4)$$

The inverse mapping function of  $\Pi$  is defined as  $\Pi^{-1}$ , which indicates the mapping from the point  $\mathbf{p}$  to the point  $P'$  on the spherical image surface as follows:

$$\mathbf{P}_c' = \Pi^{-1}(\mathbf{p}) \quad (5)$$

where  $\mathbf{P}'_c$  is the coordinate vector of point  $P'$  in the camera coordinate system. Notice that  $|\mathbf{P}'_c| = \sqrt{x^2 + y^2 + z^2} = 1$ .

#### 4. FSD-BRIEF Descriptor

The procedure of extracting the FSD-BRIEF descriptor includes four steps, namely, pixel density function designing, 3D gray centroid calculation, feature point attitude matrix construction, and FSD-BRIEF descriptor extraction. In the spherical perspective model, the densities of pixels are distributed unevenly, lowering the effectiveness of descriptors. Thus, a pixel density function is proposed firstly to calculate the distribution compensation of each pixel so as to reduce the effect of uneven pixel distribution. Then, with the help of the pixel density function, a more accurate 3D gray centroid is designed to determine the direction of FSD-BRIEF descriptor and keep its rotation invariance in the spherical perspective model. Next, we further devise a nonsingular form, a feature point attitude matrix, to represent the position and the direction of a feature point. Finally, based on the feature point attitude matrix, an FSD-BRIEF descriptor is extracted by a constructed coordinate mapping relation between the BRIEF template and the raw fisheye image.

##### 4.1. Pixel Density Function Designing

In this section, by defining the pixel density function, the distribution density of pixels on the unit sphere surface is expressed numerically.

Assuming that a pixel  $\mathbf{p}$  in a fisheye image occupies a small patch  $PIX\_PATCH(\mathbf{p})$  of the corresponding unit sphere, the mathematical expression of  $PIX\_PATCH(\mathbf{p})$  is given by:

$$PIX\_PATCH(\mathbf{p}) = \left\{ \Pi^{-1}(\mathbf{p} + \Delta\mathbf{p}) \mid \Delta\mathbf{p} = [\Delta u \quad \Delta v]^T, -\frac{1}{2} < \Delta u < \frac{1}{2}, -\frac{1}{2} < \Delta v < \frac{1}{2} \right\} \quad (6)$$

where  $\Delta u$  and  $\Delta v$  are the coordinate offsets under the pixel coordinate system in the fisheye image. It is obvious that the area of the patch  $PIX\_PATCH(\mathbf{p})$  will be smaller if the distance between point  $\mathbf{p}$  and its adjacent pixels is closer, which means that the pixel density of point  $\mathbf{p}$  is denser.

Therefore, the pixel density function  $m(\mathbf{p})$  is defined as the area of the patch  $PIX\_PATCH(\mathbf{p})$ . To simplify the computation of the curved surface area, we assume that the patch size is small enough to approximate as a parallelogram, so the pixel density function compensation  $m(\mathbf{p})$  can be computed by:

$$m(\mathbf{p}) = \begin{cases} \frac{1}{4} \left\| \left[ \Pi^{-1}(\mathbf{p} + \Delta\mathbf{x}) - \Pi^{-1}(\mathbf{p} - \Delta\mathbf{x}) \right] \times \left[ \Pi^{-1}(\mathbf{p} + \Delta\mathbf{y}) - \Pi^{-1}(\mathbf{p} - \Delta\mathbf{y}) \right] \right\|_2, \mathbf{p} \in I \\ 0, \mathbf{p} \notin I \end{cases} \quad (7)$$

where  $\|\cdot\|_2$  means L2 norm operation, and  $\Delta\mathbf{x}, \Delta\mathbf{y}$  are the coordinate offsets as follows:

$$\begin{aligned} \Delta\mathbf{x} &= [1 \quad 0]^T \\ \Delta\mathbf{y} &= [0 \quad 1]^T \end{aligned} \quad (8)$$

From Equation (7), the pixel density function  $m(\mathbf{p})$  of the whole FoV only depends on the mapping function  $\Pi$  of the spherical perspective model in Equation (4).

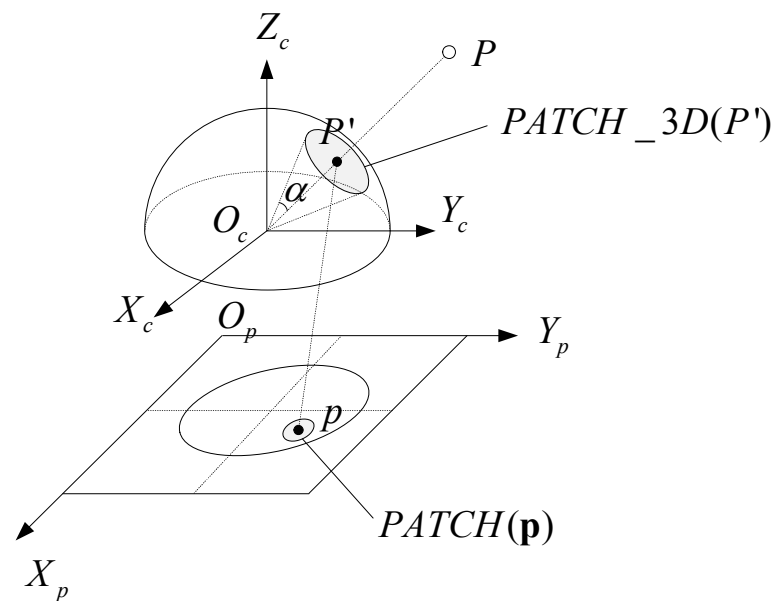
##### 4.2. 3D Gray Centroid Calculation

To determine the direction of the FSD-BRIEF descriptor, we propose a 3D gray centroid. Compared with 2D gray centroid [13,14,26], the proposed 3D gray centroid is more accurate since it takes full advantage of the consistent scale factor on the spherical perspective model. The 3D gray centroid is calculated in a circle area on the unit spherical surface. Figure 2 illustrates the correspondence of the circle area between the unit spherical surface and the fisheye image plane. As shown in Figure 2, for a FAST (Features From Accelerated Segment Test) [29] feature point  $\mathbf{p}$ , its projection point on the unit spherical surface is  $P'$ , and its

3D gray centroid calculation area is the circle area  $PATCH\_3D(P')$  with  $P'$  as the center.  $PATCH(\mathbf{p})$  is the projection of the  $PATCH\_3D(P')$  in the fisheye image plane  $O_p X_p Y_p$ .  $\alpha$  is half of the apex angle of the cone formed by  $PATCH\_3D(P')$  and the origin point  $O_c$ .

Note that the horizontal and vertical angular resolutions of fisheye cameras are approximately  $f_x$  and  $f_y$  (Pixels Per Radian) in KB4 model, and the values of  $f_x$  and  $f_y$  are often very close. In order to make the radius of the circular range cover about 15 pixel width while ensure the same mathematical status of  $f_x$  and  $f_y$ , the value of  $\alpha$  in radians is selected as 15 divided by the arithmetic mean of  $f_x$  and  $f_y$ , that is,

$$\alpha = \frac{15}{\frac{f_x + f_y}{2}} = \frac{30}{f_x + f_y} \quad (9)$$



**Figure 2.** The circle area for 3D gray centroid calculation on the unit spherical surface and its projection area in the fisheye image plane.

Define the projection area  $PATCH(\mathbf{p})$  as:

$$PATCH(\mathbf{p}) = \left\{ (\mathbf{p} + \Delta\mathbf{p}) \mid \Pi^{-1}(\mathbf{p} + \Delta\mathbf{p}) \cdot \Pi^{-1}(\mathbf{p}) > \cos \alpha \right\} \quad (10)$$

where  $\Delta\mathbf{p}$  is the offset from the pixel  $\mathbf{p}$  to the pixel in the area  $PATCH(\mathbf{p})$  in the fisheye image plane.  $\Pi^{-1}(\mathbf{p})$  is the position vector of  $P'$ .  $P'$  is also the projection point of the pixel  $\mathbf{p}$  on the unit sphere.  $\Pi^{-1}(\mathbf{p} + \Delta\mathbf{p})$  represents the position vector of the projection point of the pixel  $\mathbf{p} + \Delta\mathbf{p}$  on the unit sphere.  $\Pi^{-1}(\mathbf{p} + \Delta\mathbf{p}) \cdot \Pi^{-1}(\mathbf{p}) > \cos \alpha$  means that the angle between the two vectors  $\Pi^{-1}(\mathbf{p} + \Delta\mathbf{p})$  and  $\Pi^{-1}(\mathbf{p})$  is less than  $\alpha$ . The region  $PATCH(\mathbf{p})$  is actually the projection area of the region  $PATCH\_3D(P')$  on the fisheye image.

The 3D gray centroid of the feature point  $\mathbf{p}$  is defined as  $C$ . The symbol  $C_c$  denotes the coordinate vector of  $C$  in the camera coordinate system. The calculation formula of  $C_c$  is:

$$C_c = \frac{\sum_{\hat{\mathbf{p}} \in PATCH(\mathbf{p})} \Pi^{-1}(\hat{\mathbf{p}}) m(\hat{\mathbf{p}}) I(\hat{\mathbf{p}})}{\sum_{\hat{\mathbf{p}} \in PATCH(\mathbf{p})} m(\hat{\mathbf{p}}) I(\hat{\mathbf{p}})} \quad (11)$$

where  $\hat{\mathbf{p}}$  is a pixel in  $PATCH(\mathbf{p})$ .  $I(\mathbf{p}_k)$  represents the gray value of the pixel  $\hat{\mathbf{p}}$  in  $PATCH(\mathbf{p})$ ,  $m(\hat{\mathbf{p}})$  is the pixel density function value of  $\hat{\mathbf{p}}$ ,  $\Pi^{-1}(\hat{\mathbf{p}})$  indicates the 3D coordinate of the projection point on the unit sphere surface of  $\hat{\mathbf{p}}$ .

#### 4.3. Feature Point Attitude Matrix Construction

In order to avoid the singularity of direction expression of feature points on the poles of the unit spherical surface [25], we propose a feature point attitude matrix, a nonsingular expression, to represent the position and the direction of a feature point. The feature point attitude coordinate system  $O_b X_b Y_b Z_b$  is shown in Figure 3. The origin point  $O_b$  coincides with the origin point  $O_c$  of the camera coordinate system. The Z-axis  $O_b Z_b$  coincides with the vector  $\mathbf{P}'_c$ . The Y-axis  $O_b Y_b$  is consistent with the  $\mathbf{P}'_c \times \mathbf{C}_c$ . The X-axis  $O_b X_b$  direction is determined by right-hand rule. The X-axis is coplanar with the 3D gray centroid vector  $\mathbf{C}_c$  and the position vector  $\mathbf{P}'_c$ .

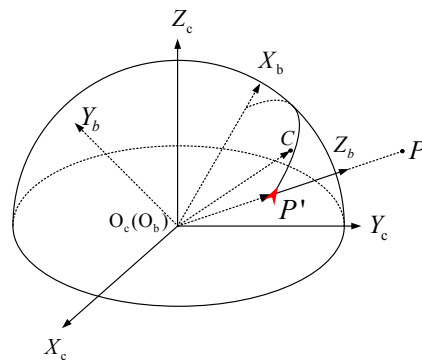


Figure 3. Feature point attitude coordinate system.

The coordinate transformation matrix  $\mathbf{R}_{cb}$  from the feature point attitude coordinate system to the camera coordinate system can be obtained as follows:

$$\mathbf{R}_{cb} = \begin{bmatrix} \mathbf{C}_c - \frac{\mathbf{C}_c \cdot \mathbf{P}'_c}{\mathbf{P}'_c \cdot \mathbf{P}'_c} \mathbf{P}'_c & \frac{\mathbf{P}'_c \times \mathbf{C}_c}{|\mathbf{P}'_c \times \mathbf{C}_c|} & \mathbf{P}'_c \end{bmatrix} \quad (12)$$

The matrix  $\mathbf{R}_{cb}$  is defined as feature point attitude matrix.

#### 4.4. FSD-BRIEF Descriptor Extraction

In this section, to enhance the distortion invariance of the descriptor in the fisheye image, FSD-BRIEF will be extracted by distorting the BRIEF template based on the constructed feature point attitude matrix so that its template can fit the distortion form of the adjacent area of the feature point.

At first, for a feature point, we define its square neighborhood region as a BRIEF template with a coordinate system  $O_B X_B Y_B$  whose origin point  $O_B$  is located at the feature point and coordinate ranges from  $-15$  to  $15$ , as shown in Figure 4. The green lines are the selected 256 groups of pixel pairs on the template.

Then, the defined BRIEF template is scaled to a certain extent and placed at the feature point as shown in Figure 5. For doing so, the following three conditions must be satisfied:

1. The center point  $O_B$  of the descriptor template coincides with the projection point  $P'$  of the feature point  $\mathbf{p}$  on the sphere. In other words, the coordinate of point  $O_B$  in the feature point attitude coordinate system is  $[0 \ 0 \ 1]^T$ .
2. The directions of  $O_B X_B$ ,  $O_B Y_B$  axis of BRIEF template coordinate system are consistent with the directions of  $O_b X_b$ ,  $O_b Y_b$  axis of the feature point attitude coordinate system.
3. There is a scale factor  $\frac{\alpha}{15}$  between the coordinates in the BRIEF template coordinate system and the coordinates in the feature point attitude coordinate system.



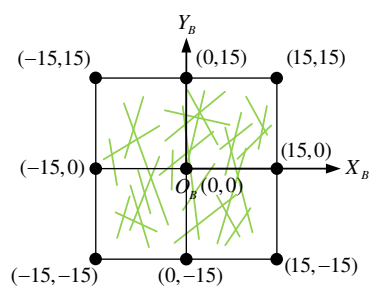


Figure 4. BRIEF template and its coordinate system.

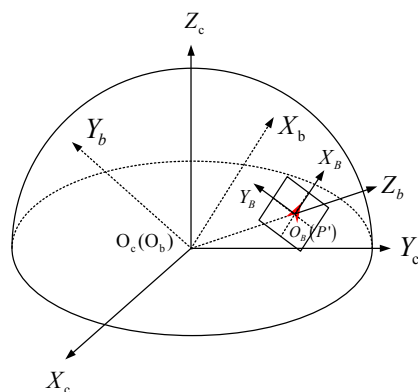


Figure 5. Position relationship between BRIEF template and spherical projection surface.

Figure 6 shows a zoom-in of a local area along the direction of  $O_b Z_b$  in Figure 5 at the feature point  $P'$ . As shown in Figure 6, for a point  $P''$  on the BRIEF template, its homogeneous coordinate vector in  $O_B X_B Y_B$  coordinate system is  $\mathbf{s}$ . The coordinate vector of point  $P''$  in the feature point attitude coordinate system is  $\mathbf{P}''_b$ . Then, the  $\mathbf{P}''_b$  can be solved by:

$$\mathbf{P}''_b = \mathbf{D}\mathbf{s} \tag{13}$$

where

$$\mathbf{D} = \text{diag}\left(\frac{\alpha}{15}, \frac{\alpha}{15}, 1\right) \tag{14}$$

$$\mathbf{s} = [s_x \quad s_y \quad 1]^T$$

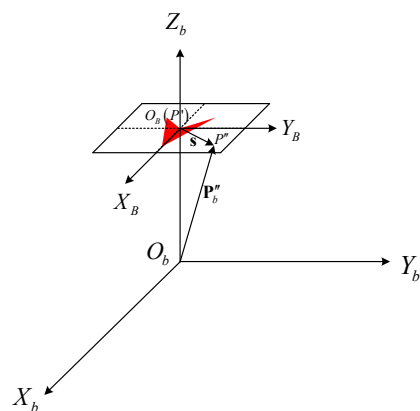


Figure 6. Coordinate mapping between BRIEF template coordinate system and feature point attitude coordinate system.

According to the law of 3D coordinate transformation and the  $\mathbf{P}_b''$ , the coordinate  $\mathbf{P}_c''$  of point  $P''$  in the camera coordinate system can be calculated by:

$$\mathbf{P}_c'' = \mathbf{R}_{cb} \mathbf{P}_b'' \quad (15)$$

where  $\mathbf{R}_{cb}$  is the feature point attitude matrix.

The projection point  $\mathbf{p}''$  of  $\mathbf{P}_c''$  in the fisheye image can be obtained by:

$$\mathbf{p}'' = \Pi(\mathbf{P}_c'') \quad (16)$$

To sum up, for a feature point whose attitude matrix is  $\mathbf{R}_{cb}$ , the coordinate mapping relationship between the point  $\mathbf{s}$  in the BRIEF template and the projection point  $\mathbf{p}''$  in the fisheye image is:

$$\mathbf{p}'' = \Pi(\mathbf{R}_{cb} \mathbf{D} \mathbf{s}) \quad (17)$$

According to Equation (17), the FSD-BRIEF of a feature point can be extracted by the calculated projection points of the FSD-BRIEF template in the fisheye image. Figure 7 shows the general view of the FSD-BRIEF descriptor. It is clear that the FSD-BRIEF template in the fisheye image changes with the position where the feature point is located, so as to ensure that the descriptor is adaptive to the different distortions in the fisheye image, and achieves a good performance on distortion invariance.

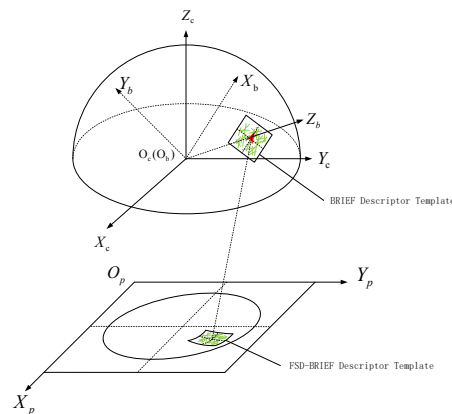


Figure 7. General view of FSD-BRIEF descriptor.

## 5. Experimental Evaluation

In this section, we present four experiments that were used for evaluating the performance of the proposed method. Experiment 1 was an ablation experiment carried out on a virtual dataset, which was used to verify the contribution of pixel density function towards improving the solution accuracy of FSD-BRIEF orientation. Experiment 2 was also conducted on the virtual dataset, aiming to prove the invariance of FSD-BRIEF compared with three BRIEF-based descriptors. Experiment 3 and Experiment 4 were performed to evaluate the matching performance of FSD-BRIEF under (1) different camera motions on a real dataset, and (2) distortion conditions on sRD-SIFT dataset [22], respectively. The results of these two experiments were compared with those produced by five state-of-the-art features.

### 5.1. Experiment 1: The Contribution Evaluation of the Pixel Density Function to the Accuracy of Feature Point Orientation

**Dataset:** In this experiment, we investigated the contribution of the pixel density function to the accuracy of feature point orientation. In order to have accurate ground truth of the direction of feature points, we produced a virtual dataset by simulating a projection of the first image of the Graffiti dataset [30]; this was used as a test image to two virtual fisheye cameras with different intrinsic parameters. At first, in the test image,

$N_p$  feature points  $\mathbf{p}_t^i (i = 1, 2, \dots, N_p)$  were extracted. During the generation of the virtual dataset, the test image and a selected virtual fisheye camera were placed in the same virtual space. By placing the test image in different poses, we projected each feature point in the fisheye image on several selected positions with different longitude angle  $\varphi$  and latitude angle  $\theta$ . The relationship between the angle  $\varphi$ ,  $\theta$  and the pose of the test image is shown in Appendix A.  $\varphi$  takes  $N_\varphi$  values and  $\theta$  takes  $N_\theta$  values. For each virtual fisheye camera,  $N_p \times N_\varphi \times N_\theta$  test samples were generated. Each test sample consisted of a generated fisheye image  $I(\varphi, \theta, \mathbf{p}_t^i)$ , a corresponding feature point position  $\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i)$  in the fisheye image, and a ground truth feature point attitude matrix  $\mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i)$ . More details of the dataset are given in Appendix B.

**Baseline:** To verify the effectiveness of the pixel density function compensation proposed in this paper, we compared two algorithms, namely, the feature point attitude matrix computation part of FSD-BRIEF without the compensation (version 1) and with (version 2). In version 1, the 3D gray centroid was calculated without the pixel density compensation term  $m(\hat{\mathbf{p}})$ . That is, the gray centroid computation formula of version 1 is shown as Equation (18). In version 2, we used Equation (11) to calculate the 3D gray centroids of feature points.

$$\mathbf{C}_c = \frac{\sum_{\hat{\mathbf{p}} \in \text{PATCH}(\mathbf{p})} \mathbf{\Pi}^{-1}(\hat{\mathbf{p}}) I(\hat{\mathbf{p}})}{\sum_{\hat{\mathbf{p}} \in \text{PATCH}(\mathbf{p})} I(\hat{\mathbf{p}})} \quad (18)$$

**Fisheye cameras:** In order to verify the contribution of the pixel density function under different FoV cameras, two virtual cameras with different FoVs were selected for this experiment. Table 1 shows the intrinsic parameters of the two cameras.

**Table 1.** The Intrinsic Parameters of 170° FoV and 210° FoV Camera.

Intrinsic Parameter	170° FoV Camera	210° FoV Camera
$f_x$	284.977	257.28
$f_y$	284.977	257.28
$c_x$	423.039	582.006
$c_y$	398.179	419.655
$k_1$	−0.00454	−0.0765
$k_2$	0.0396	0.00908
$k_3$	−0.0363	−0.0117
$k_4$	0.00584	0.00373

Figure 8 shows the curve of the pixel density function of 170° FoV and 210° FoV cameras with  $\theta$ . From the curve, we can see that the curve of the pixel density function of 170° FoV cameras decreased in angle range 0–60°, and increased in angle range 60–80°. Another curve, which was for the pixel density function of 210° FoV camera, increased in the whole angle range of 0–90°.

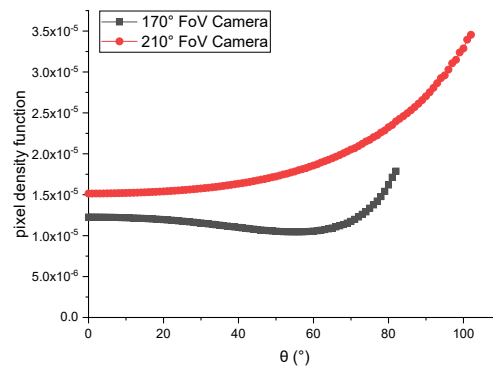
**Evaluation metrics:** In the experimental verification process, the direction angle error of the feature point is used for quantitative evaluation. The direction angle error, denoted by  $e(\varphi, \theta, \mathbf{p}_t^i)$ , is shown in Figure 9, where  $P^i(\varphi, \theta, \mathbf{p}_t^i)$  is the projection point of  $\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i)$  on the unit sphere surface. The coordinate system  $O_b^* X_b^* Y_b^* Z_b^*$  is the feature point attitude coordinate system corresponding to the ground truth feature point attitude matrix  $\mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i)$ , whilst  $O_b X_b Y_b Z_b$  is the feature point attitude coordinate system corresponding to the calculated feature point attitude matrix  $\mathbf{R}_{cb}^i(\varphi, \theta, \mathbf{p}_t^i)$ . Note that  $\overrightarrow{O_b^* X_b^*}$  and  $\overrightarrow{O_b X_b}$  are defined as the ground truth direction and the calculated direction of the feature point (see Section 4.3). The unit of  $e(\varphi, \theta, \mathbf{p}_t^i)$  is defined as degree (°). Let  $(\overrightarrow{O_b^* X_b^*})_c$

and  $(\overrightarrow{O_b X_b^*})_c$  be the coordinate of the unit direction vectors corresponding to  $\overrightarrow{O_b^* X_b^*}$  and  $\overrightarrow{O_b X_b}$  in the camera coordinate system, then:

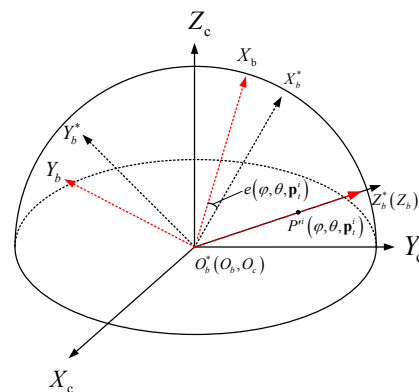
$$\begin{aligned} (\overrightarrow{O_b^* X_b^*})_c &= \mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i) [1 \ 0 \ 0]^T \\ (\overrightarrow{O_b X_b})_c &= \mathbf{R}_{cb}^i(\varphi, \theta, \mathbf{p}_t^i) [1 \ 0 \ 0]^T \\ \cos\left(\frac{\pi}{180} e(\varphi, \theta, \mathbf{p}_t^i)\right) &= \overrightarrow{O_b X_b} \cdot \overrightarrow{O_b^* X_b^*} = (\overrightarrow{O_b X_b})_c^T (\overrightarrow{O_b^* X_b^*})_c \end{aligned} \quad (19)$$

From Equation (19), we can obtain the expression of  $e(\varphi, \theta, \mathbf{p}_t^i)$  as:

$$e(\varphi, \theta, \mathbf{p}_t^i) = \frac{180}{\pi} \arccos\left([1 \ 0 \ 0] \mathbf{R}_{cb}^i(\varphi, \theta, \mathbf{p}_t^i) \mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i) [1 \ 0 \ 0]^T\right) \quad (20)$$



**Figure 8.** The pixel density function of 170° field of view (FoV) and 210° FoV camera with  $\theta$ .



**Figure 9.** The definition of feature point direction angle error between calculated direction and ground truth direction.

Note that values of  $e(\varphi, \theta, \mathbf{p}_t^i)$  could be calculated from experimental results indexed by  $\varphi$  (FoV longitude angle),  $\theta$  (FoV Latitude Angle) and  $i$  (feature point index in test image). For an ideal method,  $e(\varphi, \theta, \mathbf{p}_t^i)$  is always zero, and the calculated direction of feature point is consistent with the real direction. In fact, due to the influence of noise, the angle error  $e(\varphi, \theta, \mathbf{p}_t^i)$  would not be zero. In this experiment, the smaller the value of  $e(\varphi, \theta, \mathbf{p}_t^i)$ , the more accurate the calculated feature point direction.

In this study, the mean error  $e_{mean}(\theta)$  and the standard deviation  $e_{SD}(\theta)$  were used to evaluate the results of  $e(\varphi, \theta, \mathbf{p}_t^i)$ .  $e_{mean}(\theta)$  measures the average error of the feature point direction calculated by using all the points under the latitude angle  $\theta$ .  $e_{SD}(\theta)$  measures the dispersion of the  $e(\varphi, \theta, \mathbf{p}_t^i)$  distribution under  $\theta$ .  $e_{mean}(\theta)$  and  $e_{SD}(\theta)$  are calculated as follows:

$$e_{mean}(\theta) = \frac{\sum_{\varphi} \sum_i e(\varphi, \theta, \mathbf{p}_t^i)}{N_{\varphi} N_p}$$

$$e_{SD}(\theta) = \sqrt{\frac{\sum_{\varphi} \sum_i [e(\varphi, \theta, \mathbf{p}_t^i) - e_{mean}(\theta)]^2}{N_{\varphi} N_p}}$$
(21)

where, the  $N_{\varphi}$  and  $N_p$  are the number of  $\varphi$  and  $i$  values. The smaller the  $e_{mean}(\theta)$  is, the more accurate the feature point direction is. The smaller the  $e_{SD}(\theta)$  is, the more stable the result of feature point direction is.

**Evaluations:** In the 170° FoV camera, the range of  $\theta$  is 10–80°. In the 210° FoV camera, the range of  $\theta$  is 10–90°. The two statistics  $e_{mean}(\theta)$  and  $e_{SD}(\theta)$  are computed for both of the two cases. The comparison results are shown in Tables 2 and 3. The error reduction of version 2 compared to version 1 are calculated as follows:

$$\eta = \frac{e_{v2} - e_{v1}}{e_{v1}} \times 100\%$$
(22)

where  $\eta$  is the value of error reduction,  $e_{v1}$  and  $e_{v2}$  are the value of the direction angle error of version 1 and version 2 individually. Taking the horizontal axis as the  $\theta$  value and the vertical axis as  $e_{mean}(\theta)$  and  $e_{SD}(\theta)$ , the  $e - \theta$  curves are also drawn in Figure 10.

**Table 2.** The numerical results of direction angle error in 170° FoV camera.

$\theta$ (°)	Version 1 (Without Compensation)		Version 2 (With Compensation)		Error Reduction (%)	
	Mean	SD	Mean	SD	Mean	SD
10	1.133	0.920	1.084	0.865	−4.306	−5.978
20	1.213	0.827	1.162	0.800	−4.140	−3.360
30	1.034	0.786	0.922	0.703	−10.895	−10.452
40	1.143	0.914	0.948	0.782	−17.111	−14.451
50	1.106	0.905	1.116	0.811	0.895	−10.367
60	1.030	0.796	0.947	0.668	−8.033	−16.065
70	1.756	1.251	0.849	0.656	−51.629	−47.526
80	5.185	3.326	1.342	1.011	−74.123	−69.592

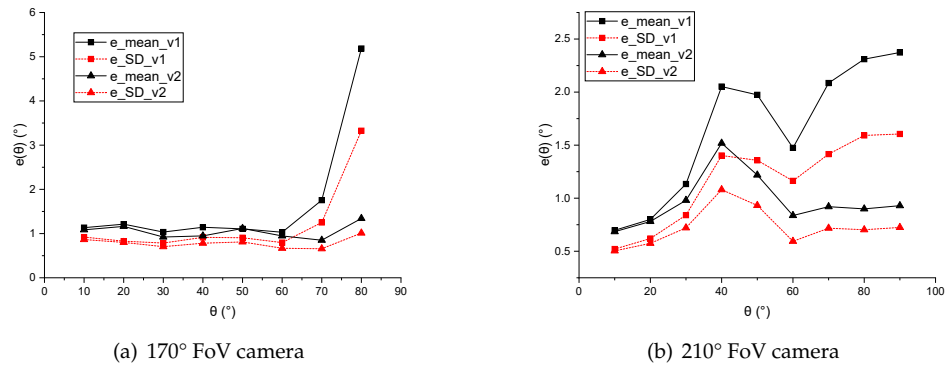
**Table 3.** The numerical results of direction angle error in 210° FoV camera.

$\theta$ (°)	Version 1 (Without Compensation)		Version 2 (With Compensation)		Error Reduction (%)	
	Mean	SD	Mean	SD	Mean	SD
10	0.697	0.521	0.684	0.504	−1.850	−3.322
20	0.800	0.620	0.781	0.574	−2.425	−7.339
30	1.134	0.840	0.980	0.720	−13.540	−14.277
40	2.052	1.402	1.518	1.080	−25.995	−22.989
50	1.974	1.357	1.218	0.932	−38.280	−31.344
60	1.474	1.163	0.837	0.594	−43.226	−48.942
70	2.085	1.415	0.920	0.717	−55.880	−49.322
80	2.310	1.591	0.899	0.703	−61.068	−55.838
90	2.373	1.605	0.929	0.725	−60.838	−54.859

For the 170° FoV camera, both of the two compensation schemes led to similarly stable results in the angle range of 10–60°. However, when the angle  $\theta$  became large (especially in the range of 60–80°), the performance of Version 2 was obviously much better than that of Version 1. Both of the average angle error and the accuracy dispersion of the proposed method (version 2) were about 1° in the whole fisheye FoV of the dataset.

For the 210° FoV camera, the overall performance of Version 2 was continuously better than that of Version 1 throughout the range of 30–90°.

The experimental results showed that near the edge of FoV, especially in the FoV region where the pixel density function increased monotonously with the angle  $\theta$ , the pixel density compensation improved the accuracy and stability of feature point direction calculation significantly.



**Figure 10.**  $e - \theta$  curves of two versions of feature point direction calculation methods in 170° FoV camera and 210° FoV camera.

### 5.2. Experiment 2: Descriptor Invariance Evaluation of Fisheye Images in Different FoV Positions

**Baselines:** In this experiment, three typical BRIEF descriptors, including ORB, dBRIEF (Distorted BRIEF), and mdBRIEF, were selected as baselines. The descriptor of the feature point in each test sample in the virtual dataset generated in Experiment 1 was extracted by the tested features (FSD-BRIEF, ORB, dBRIEF, and mdBRIEF). In order to ensure a fair comparison of experimental results, all the binary descriptors were chosen to be 256 bits. dBRIEF is the version of mdBRIEF without on-line mask learning. For dBRIEF and mdBRIEF, we used the open source version provided in GitHub. For ORB, we used the functions provided in OpenCV and its default parameter settings.

**Evaluation metrics:** In this experiment, we define  $D(\varphi, \theta, \mathbf{p}_t^i)$  as the descriptor of the feature point  $\mathbf{p}_t^i(\varphi, \theta, \mathbf{p}_t^i)$ . The associated Hamming distance error  $\Delta D(\varphi, \theta, \mathbf{p}_t^i)$  of the descriptor of the feature point was used to evaluate the invariance performance of algorithms.  $\Delta D(\varphi, \theta, \mathbf{p}_t^i)$  is calculated for each feature point test sample by each test feature as:

$$\Delta D(\varphi, \theta, \mathbf{p}_t^i) = h(D(\varphi, \theta, \mathbf{p}_t^i), D(\varphi_0, \theta_0, \mathbf{p}_t^i)) \quad (23)$$

here we selected  $D(\varphi_0, \theta_0, \mathbf{p}_t^i)$  as the reference standard descriptor to compute the Hamming distance error, where  $\varphi_0 = 45^\circ$   $\theta_0 = 10^\circ$ . For an ideal feature algorithm, for the same  $\mathbf{p}_t^i$ , no matter what values of  $\varphi$  and  $\theta$  take, there is  $\Delta D(\varphi, \theta, \mathbf{p}_t^i) = 0$ . However, in practice, due to the resampling error of the fisheye camera,  $\Delta D(\varphi, \theta, \mathbf{p}_t^i)$  was not zero. Therefore, the smaller the calculated value of  $\Delta D(\varphi, \theta, \mathbf{p}_t^i)$ , the stronger the invariance of the feature algorithm to radial distortion of the fisheye image.

Similar to Experiment 1,  $\Delta D_{mean}(\theta)$  and  $\Delta D_{SD}(\theta)$  were used as evaluation metrics.  $\Delta D_{mean}(\theta)$  is the average value of the descriptor distance calculated by using all the points under the latitude angle  $\theta$ .  $\Delta D_{SD}(\theta)$  is the dispersion of the  $\Delta D(\varphi, \theta, \mathbf{p}_t^i)$  distribution under  $\theta$ . The smaller the  $\Delta D_{mean}(\theta)$  is, the stronger the invariance of feature algorithm to radial distortion of fisheye images. The smaller the  $\Delta D_{SD}(\theta)$  is, the more stable the performance of the feature algorithm is. The computation formula of  $\Delta D_{mean}(\theta)$  and  $\Delta D_{SD}(\theta)$  was as follows:

$$\Delta D_{mean}(\theta) = \frac{\sum_{\varphi} \sum_i \Delta D(\varphi, \theta, \mathbf{p}_t^i)}{N_{\varphi} N_{\mathbf{p}}}$$

$$\Delta D_{SD}(\theta) = \sqrt{\frac{\sum_{\varphi} \sum_i [\Delta D(\varphi, \theta, \mathbf{p}_t^i) - \Delta D_{mean}(\theta)]^2}{N_{\varphi} N_{\mathbf{p}}}}$$
(24)

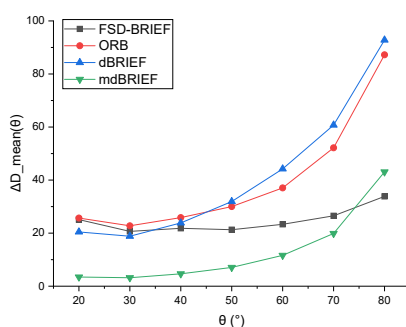
**Evaluations:** Since  $\theta_0 = 10^\circ$  was set for the reference standard descriptor  $D(\varphi_0, \theta_0, \mathbf{p}_t^i)$ , the ranges of  $\theta$  were selected as  $20\text{--}80^\circ$  in  $170^\circ$  FoV camera, and  $20\text{--}90^\circ$  in  $210^\circ$  FoV camera respectively. The values of  $\Delta D_{mean}(\theta)$  and  $\Delta D_{SD}(\theta)$  of FSD-BRIEF, ORB, dBRIEF, and mdBRIEF were computed. The numerical results are shown in Tables 4 and 5. The corresponding curves of  $\Delta D_{mean}(\theta)$  are shown in Figure 11, and the curves of  $\Delta D_{SD}(\theta)$  are shown in Figure 12.

**Table 4.** The numerical results of Hamming distance error in  $170^\circ$  FoV camera.

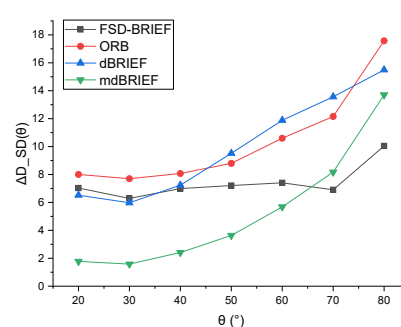
$\theta$ ( $^\circ$ )	FSD-BRIEF		ORB		dBRIEF		mdBRIEF	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
20	25.100	7.033	25.692	8.005	20.458	6.520	3.458	1.779
30	20.658	6.284	22.767	7.700	18.833	5.974	3.192	1.583
40	21.825	6.994	25.867	8.073	23.850	7.237	4.667	2.413
50	21.300	7.209	30.017	8.798	31.917	9.516	7.083	3.635
60	23.325	7.407	37.050	10.598	44.217	11.882	11.633	5.680
70	26.533	6.904	52.175	12.154	60.742	13.563	19.883	8.170
80	33.850	10.045	87.233	17.572	92.792	15.495	43.083	13.704

**Table 5.** The numerical results of Hamming distance error in  $210^\circ$  FoV camera.

$\theta$ ( $^\circ$ )	FSD-BRIEF		ORB		dBRIEF		mdBRIEF	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
20	20.892	5.639	21.158	6.309	17.600	4.467	2.775	1.345
30	22.608	6.125	25.467	7.242	20.000	5.804	3.208	1.460
40	25.767	7.475	30.925	8.592	23.708	7.439	3.792	1.788
50	25.875	7.996	41.442	11.538	31.083	9.718	5.058	1.881
60	28.867	7.978	57.508	15.822	42.558	13.937	8.058	4.101
70	30.317	8.176	70.775	16.628	58.758	14.927	17.892	8.579
80	36.250	10.375	84.217	16.844	95.292	16.522	44.975	14.635
90	45.000	14.170	97.450	21.129	-	-	-	-



(a)  $\Delta D_{mean}(\theta)$  Curves



(b)  $\Delta D_{SD}(\theta)$  Curves

**Figure 11.**  $\Delta D - \theta$  curve results in  $170^\circ$  FoV camera.

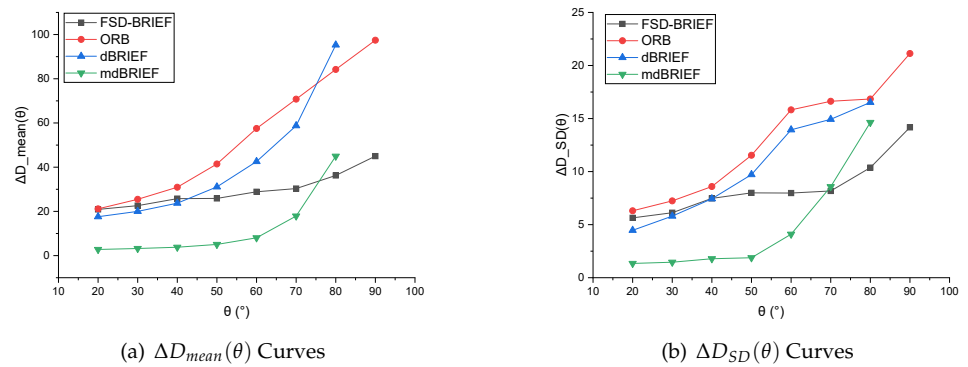


Figure 12.  $\Delta D - \theta$  curve results in 210° FoV camera.

The experimental results of the two cameras showed that, in the angle range of 20°–40°, FSD-BRIEF led to similarly stable descriptor errors as ORB and dBRIEF. However, in the angle range of 40°–80°, the descriptor errors of ORB and dBRIEF tended to increase significantly, while the descriptor errors of FSD-BRIEF increased much less than that of ORB and dBRIEF. In the angle range of 75°–80°, the descriptor error of FSD-BRIEF was smaller than that of mdBRIEF. However, the descriptor error of FSD-BRIEF was larger than that of mdBRIEF in the angle range of 20°–60°; this is because an on-line mask learning scheme was performed in mdBRIEF, where the unstable binary bits were masked.

The standard deviations (SD) of FSD-BRIEF, ORB and dBRIEF were similar in the angle range of 20°–40°. In the angle range up to 50°, the SD of FSD-BRIEF was significantly smaller than that of ORB and dBRIEF. In the angle range of 20°–60°, the SD value of FSD-BRIEF was not as small as mdBRIEF, but smaller than mdBRIEF in the angle range of 70°–80°.

Because dBRIEF and mdBRIEF distorted the descriptor template based on the plane perspective model, it could not extract the feature descriptor when  $\theta$  was 90°, and there was no 90° effective value of the descriptor errors.

It can be observed from the results that, compared with other BRIEF based features, FSD-BRIEF could effectively adapt to the radial distortion of fisheye images and ensure the invariance of descriptors.

### 5.3. Experiment 3: Matching Performance Evaluation in Different Kind of Image Variance

**Dataset:** In order to verify the FoV edge distortion invariance, translation invariance, and scale invariance performances of the proposed FSD-BRIEF in image matching process, a dataset captured by a 210° FoV fisheye camera was made. The intrinsic parameter of the 210° FoV fisheye camera is shown in Table 1. There were three groups of images in this dataset, and each group contained 13 images. In the first group of images, through rotation of the camera, the test image fell on the edge of the camera’s FoV as close as possible, and the test image was distorted by the radial distortion of the fisheye camera to the greatest extent. In the second group of images, by moving and rotating the camera parallel to the test image plane, the test image fell in different positions of the camera FoV. In the third group of images, the camera moved forward and backward greatly relative to the test image, which made the projection of the test image in the fisheye image has a large-scale change.

**Baselines:** In this experiment, five state-of-the-art descriptors, AKAZE, BRISK, ORB, dBRIEF and mdBRIEF, were selected as baselines. For FSD-BRIEF, we used the FAST feature to extract feature points. For BRISK, ORB, and AKAZE, we used the functions provided in OpenCV with default parameter settings. For dBRIEF and mdBRIEF, we used the open source version provided in GitHub.

**Evaluation metrics:** In order to evaluate the matching performance of FSD-BRIEF proposed in this paper, according to [30], we conducted comparison experiments with state-of-the-art descriptors by calculating the PR (recall—“1-precision”) curve of the matching



results. Designate  $S^i, S^j$  to be a set of feature points detected in the image  $I^i$  and  $I^j$  respectively, then the set of ground truth matching points  $G^{ij}$  can be given by:

$$G^{ij} = \{(\mathbf{p}^i, \mathbf{p}^j) \mid \|\mathbf{p}^i - \Pi(\mathbf{H}^{ij}\Pi^{-1}(\mathbf{p}^j))\| < \varepsilon, \mathbf{p}^i \in S^i, \mathbf{p}^j \in S^j\} \quad (25)$$

where  $\|\cdot\|$  refers to Euclidean distance between the  $\mathbf{p}^i$  and the projecting point of  $\mathbf{p}^j$  in image  $I^i$ ,  $\mathbf{H}^{ij}$  is the ground truth homography matrix from image  $I^i$  to  $I^j$ , which was calculated by manually labeled corresponding points in the image sequence. The distance threshold  $\varepsilon$  was taken as 3 pixels. To evaluate the matching performance of test features, let  $M^{ij}$  be the set of matching feature point pairs gained by the algorithm from the image  $I^i$  and  $I^j$ , and  $M^{ij}$  consisted of correct matches  $M_{true}^{ij}$  and incorrect matches  $M_{false}^{ij}$ . Hence, as shown in Equation (26), the  $recall(\varepsilon')$  presents the ability of the matching algorithm to find correct matches, and  $1 - precision(\varepsilon')$  indicates the algorithm's capability of discarding unmatched points.

$$recall(\varepsilon') = \frac{\sum_{1 \leq i < j \leq n} N(M_{true}^{ij}(\varepsilon'))}{\sum_{1 \leq i < j \leq n} N(G^{ij})}, 1 - precision(\varepsilon') = 1 - \frac{\sum_{1 \leq i < j \leq n} N(M_{false}^{ij}(\varepsilon'))}{\sum_{1 \leq i < j \leq n} N(M^{ij}(\varepsilon'))} \quad (26)$$

where  $n$  is the number of images in the image sequence,  $N(*)$  denotes the point pair number of a set,  $\varepsilon'$  is a descriptor distance threshold that was used to obtain the correct matches whose Euclidean distance between their descriptors is below  $\varepsilon'$ . Each of the two measures yielded a so-called PR curve by increasing the threshold  $\varepsilon'$  from zero gradually. That PR curve passed at a short distance of the ideal point (0, 1) meant the corresponding test feature was absolutely perfect which could make both the value of precision and recall rate 1. In practice, a good matching performance was achieved when the matching algorithm's PR curve had the minimum distance to the point (0, 1), the highest recall, and the minimum 1-precision.

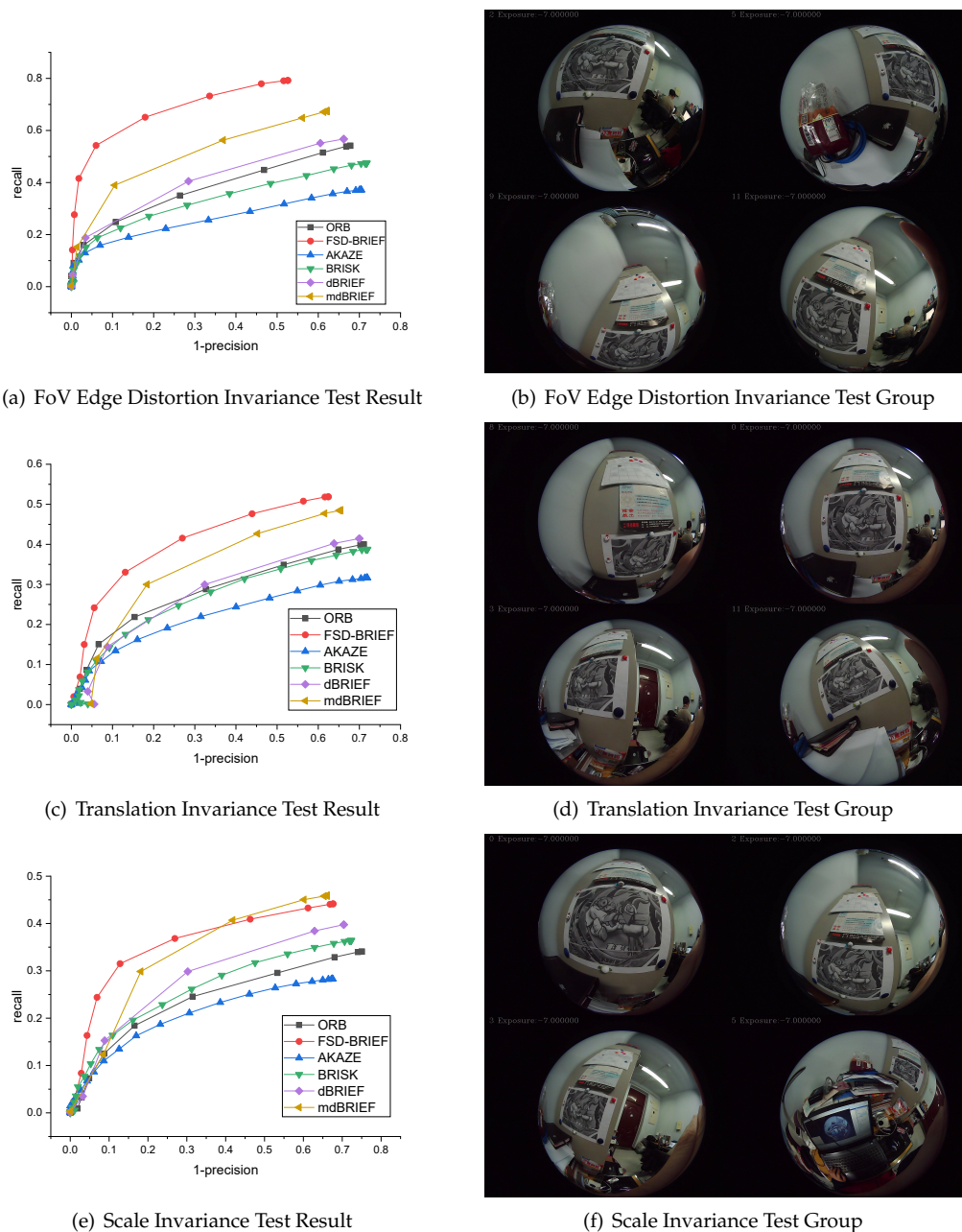
**Evaluation:** To test the matching performance in this dataset, we used the test features to extract and match features and drew PR curves. For each algorithm in each image, 300 strongest feature points were extracted. The PR curve results are shown in Figure 13.

From Figure 13a,b, the recall value at the end of the PR curve of FSD-BRIEF proposed in this paper was in the range of 0.75–0.8. For other features involved in the comparison, the recall value at the end of the PR curve was in the range of 0.3–0.6. The result showed that, compared with other features, FSD-BRIEF had significant FoV edge distortion invariance in the feature matching process of severely distorted images.

Figure 13c,d shows that the recall value at the end of the PR curve of FSD-BRIEF proposed in this paper was near 0.5. For other features involved in comparison, the recall value at the end of the PR curve was in the range of 0.25–0.5 and below FSD-BRIEF. The result showed that, compared with other features, FSD-BRIEF had better translation invariance in the feature matching process of fisheye images.

In Figure 13e,f, it can be observed that the recall value at the end of the PR curve of FSD-BRIEF proposed in this paper was in the range of 0.4–0.45. For AKAZE, BRISK, ORB, and dBRIEF, the recall value at the end of the PR curve was in the range of 0.25–0.4. The recall value of FSD-BRIEF was higher than mdBRIEF when  $1 - precision$  was in the range of 0.05–0.3. The results showed that FSD-BRIEF had better scale invariance in the feature matching process of fisheye images compared with most of the state-of-the-art features.

Using AKAZE, BRISK, ORB, dBRIEF and mdBRIEF as references, experimental results showed that FSD-BRIEF showed comparable performance in FoV edge distortion invariance, translation invariance, scale invariance, and matching performance in fisheye images.



**Figure 13.** 210° FoV camera dataset and corresponding PR curve result.

#### 5.4. Experiment 4: Matching Performance Evaluation in Different Distortion Images

**Dataset:** In order to verify the matching performance of FSD-BRIEF under different radial distortion, the sRD-SIFT dataset was used in this experiment. The sRD-SIFT datasets [22] were published with the work of sRD-SIFT. It consisted of three sets of images (FireWire, Dragonfly, and Fisheye), each set containing 13 images and captured by a camera with different radial distortion. The dataset contained significant scaling and rotation changes. Four images selected randomly for each dataset are shown in the right panels of Figure 14.

**Fisheye cameras:** The three sets of images were attached with the image of a checkerboard calibration board for camera calibration. Therefore, we calibrated each camera based on the KB4 fisheye camera model using the chessboard image provided. The calibration results are shown in Table 6.

**Table 6.** The intrinsic parameters of the cameras in sRD-SIFT datasets.

Intrinsic Parameter	set1(FireWire)	set2(Dragonfly)	set3(Fisheye)
$f_x$	539.389	528.626	306.780
$f_y$	539.389	528.626	306.780
$c_x$	312.103	365.029	634.729
$c_y$	233.050	228.558	478.546
$k_1$	0.0537	−0.0994	−0.000788
$k_2$	0.0871	−0.0205	0.0181
$k_3$	0	0.00661	−0.0117
$k_4$	0	0.0150	0.00190

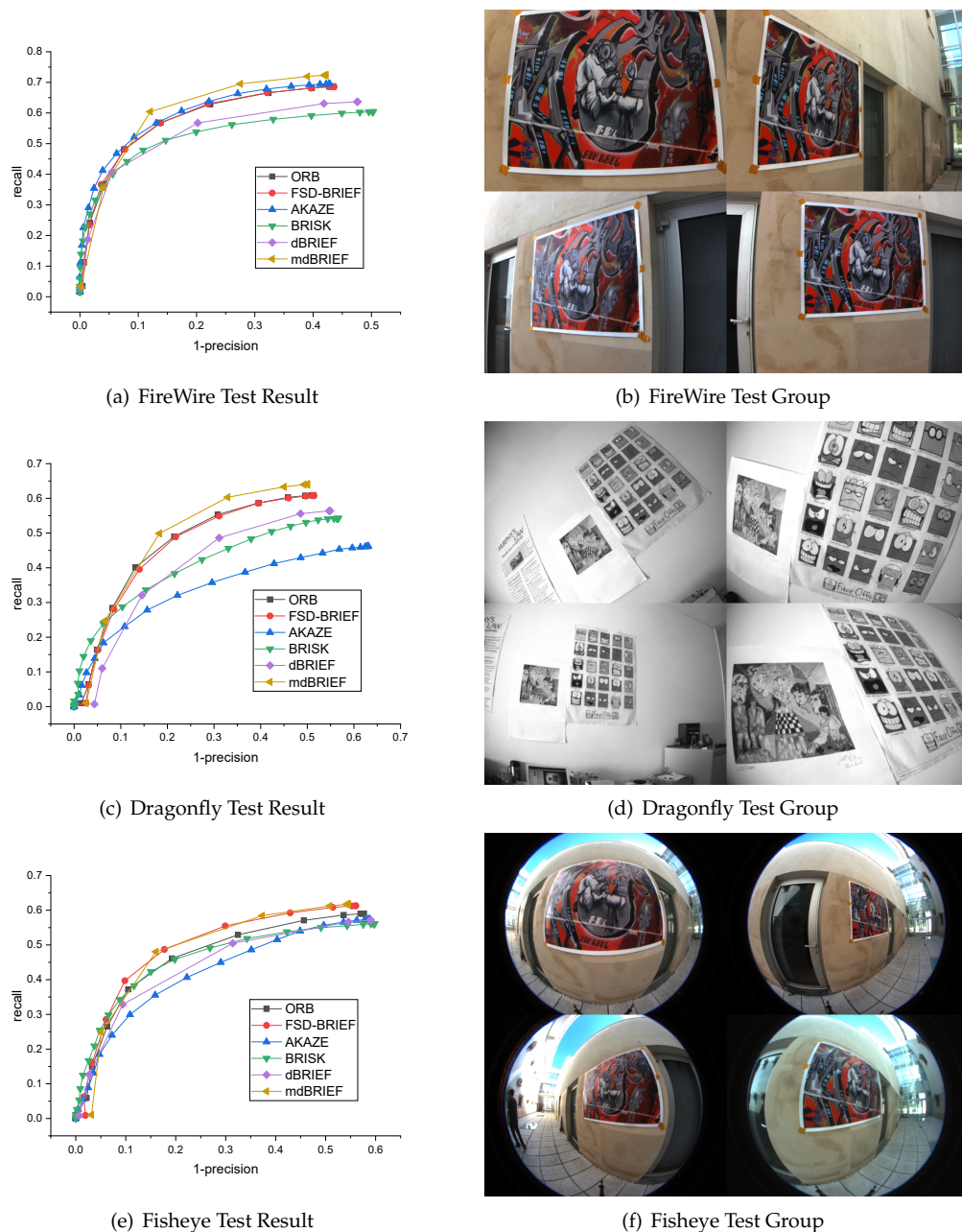
**Evaluation:** Similar to Experiment 3, to test the matching performance in the three groups of the sRD-SIFT dataset, we also employed the baseline descriptors (ORB, AKAZE, BRISK, dBRIEF and mdBRIEF) to extract and match 300 strongest keypoints for each image, then draw PR curves. The results are shown in Figure 14, where Figure 14a,b shows the results and the image group with the least distortion. Figure 14c,d shows the results and the image group with moderate distortion. Figure 14e,f shows the results of the image group with the most distortion captured by fisheye cameras.

Figure 14a,b shows that the PR curve of FSD-BRIEF almost coincided with that of ORB and AKAZE, and the performance of AKAZE was slightly better. The recall rate at the end of the curve of FSD-BRIEF, ORB, and AKAZE was in the range of 0.65–0.7, which was higher than that of BRISK and dBRIEF. From the result, we can see that the performance of FSD-BRIEF was equivalent to that of ORB in small distorted images.

Figure 14c,d shows that the PR curve of FSD-BRIEF almost coincided with that of ORB, and the recall at the end of the curve was around 0.6, which was higher than that of AKAZE, BRISK, and dBRIEF. From the result, we can see that the performance of FSD-BRIEF was equivalent to that of ORB in moderate distorted images and better than AKAZE, BRISK, and dBRIEF.

In Figure 14e,f, it can be observed that the recall value at the end of the PR curve of FSD-BRIEF was around 0.6, which was higher than that of ORB, AKAZE, BRISK, and dBRIEF, and almost the same as that of mdBRIEF. From the result, we can see that the performance of FSD-BRIEF was almost equivalent to that of mdBRIEF and better than ORB, AKAZE, BRISK, and dBRIEF in the most distorted images.

These experimental results show that the performance of FSD-BRIEF in large distortion image was better than most of the state-of-the-art features involved in the comparison. In small and moderate distorted images, the performance of FSD-BRIEF was similar to that of the ORB feature. That is because that the test image was close to the center of the FoV in this dataset, the radial distortion effect of the test image by the fisheye lens was limited compared with Experiment 3. Therefore, the performance of FSD-BRIEF in this paper on the sRD-SIFT dataset was not as prominent as the 210° FoV camera dataset in Experiment 3.



**Figure 14.** sRD-SIFT dataset and corresponding PR curve result.

## 6. Conclusions

In this paper, to tackle the problem of the feature matching performance deterioration due to the impact of fisheye radial distortion, we proposed a novel distorted BRIEF descriptor, named FSD-BRIEF, for fisheye images based on the spherical projection model. First, for reducing the impact of the distortion on gray centroid calculation and the accuracy of feature point direction, we designed a pixel density function and evaluated its performance by comparing the feature point direction error results of the algorithms with and without using the function. The obtained results shown that the pixel density function can promote the precision of the feature point direction calculation. Second, the distortion invariance of the proposed FSD-BRIEF was verified and compared with other BRIEF based descriptors, and the associated results demonstrated that FSD-BRIEF works well for distortion invariance in different positions of fisheye images. In the matching experiments in 210° FoV camera datasets, FSD-BRIEF shown better performance for FoV edge distortion

invariance, translation invariance, and scale invariance in large distortion fisheye images. In the sRD-SIFT dataset, the FSD-BRIEF descriptor can significantly improve the matching performance for large distortion images, and meanwhile can still produce excellent results for small distortion images.

## 7. Future Work

It is known that panoramic images have been widely used today. The proposed descriptor can be adapted and potentially applied to panoramic images, with some slight modifications of the camera model and the computation method of the pixel density function, respectively. Moreover, in the future work, we will design a distorted FAST detector based on the spherical perspective model for panoramic images to extract feature points at any position including the two Polar Regions.

**Author Contributions:** conceptualization Y.Z.; investigation J.S., Y.D., and Y.Y.; methodology Y.Z. and Y.D.; project administration J.S. and Y.D.; software Y.Z.; supervision J.S. and Y.D.; validation Y.Z. and Y.Y.; visualization Y.Z. and Y.Y.; writing—original draft Y.Z., Y.D., Y.Y. and H.-L.W.; writing—review and editing Y.D., Y.Y. and H.-L.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

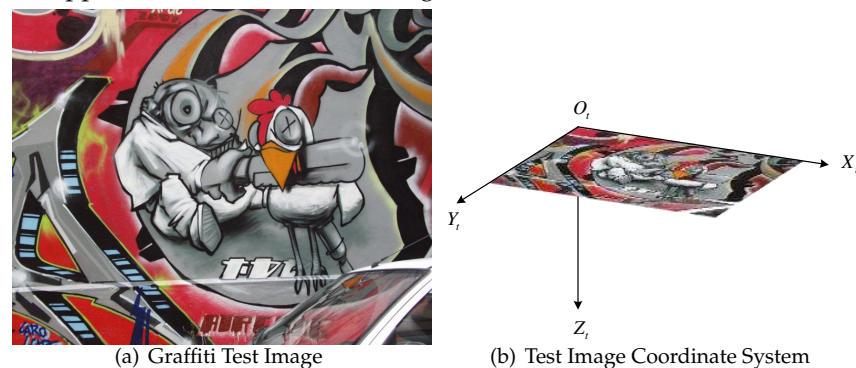
**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The datasets in Experiment 1-3 are available at <https://github.com/Ironeagleufo123/FSD-BRIEF-Dataset> (accessed on 6 March 2021). The datasets in Experiment 4 were published with the work of sRD-SIFT.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Coordinate Transformation for Virtual Dataset Generation

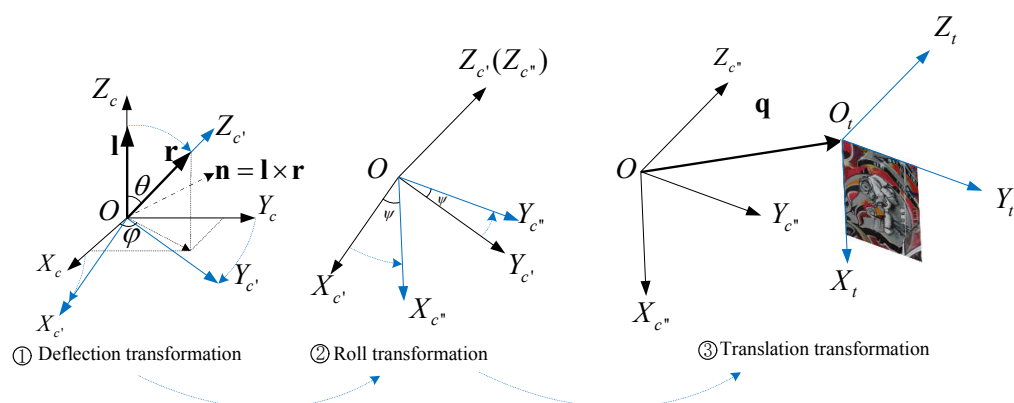
For the test picture shown in Figure A1a, define the test image coordinate system  $O_t X_t Y_t Z_t$ , as shown in Figure A1b. The coordinate origin  $O_t$  is located in the first pixel in the upper left corner of the test image.



**Figure A1.** Graffiti Test Image (a) and the Test Image Coordinate System (b).

The X-axis represents the row of the image pixel, the Y-axis represents the column of the image pixel, and the Z-axis is determined according to the right-hand rule.

In this paper, the coordinate system transformation process from the camera coordinate system  $O_c X_c Y_c Z_c$  to the test image coordinate system  $O_t X_t Y_t Z_t$  is shown in Figure A2, which mainly includes three steps: (1) Deflection transformation, (2) Roll transformation, (3) Translation transformation.



**Figure A2.** Three step transformation from the camera coordinate system to test image coordinate system.

#### Appendix A.1. Deflection Transformation

As shown on the left of Figure A2, note that the vector  $\mathbf{l}$  is the unit vector consistent with the Z-axis direction of the camera coordinate system  $O_c X_c Y_c Z_c$  and the vector  $\mathbf{r}$  is a 3D unit vector, indicating the position to which the Z-axis of the camera coordinate system will turn. The camera coordinate system is rotated around vector  $\mathbf{l} \times \mathbf{r}$  according to the right-hand rule, so that the Z-axis is consistent with the  $\mathbf{r}$  vector direction after rotation, and the transition coordinate system  $O_{c'} X_{c'} Y_{c'} Z_{c'}$  is obtained. The rotation angle is equal to the angle between vector  $\mathbf{l}$  and  $\mathbf{r}$ , which is defined as  $\theta$ .

The unit vector corresponding to the rotation axis  $\mathbf{l} \times \mathbf{r}$  is defined as  $\mathbf{n}$ , and  $\varphi$  is defined as the angle between the projection of the vector  $\mathbf{r}$  on the  $X_c O_c Y_c$  plane and the  $O_c X_c$  axis. The following constraints exist:

$$\begin{aligned} \cos \theta &= \mathbf{l} \cdot \mathbf{r}, \\ \sin \theta &= \|\mathbf{l} \times \mathbf{r}\|, \\ \mathbf{n} &= \frac{\mathbf{l} \times \mathbf{r}}{\|\mathbf{l} \times \mathbf{r}\|}, \\ \mathbf{n} &= [-\sin \varphi \quad \cos \varphi \quad 0]^T \end{aligned} \quad (\text{A1})$$

According to Rodriguez formula, the coordinate transformation matrix from the transition coordinate system  $O_{c'} X_{c'} Y_{c'} Z_{c'}$  to the camera coordinate system  $O_c X_c Y_c Z_c$  is as follows:

$$\mathbf{R}_{cc'} = \cos \theta \mathbf{I} + (1 - \cos \theta) \mathbf{nn}^T + \sin \theta \hat{\mathbf{n}} \quad (\text{A2})$$

where the symbol  $\hat{\cdot}$  represents the transformation from three-dimensional column vector to skew-symmetric matrix:

$$\left( \begin{bmatrix} x \\ y \\ z \end{bmatrix} \right)^\wedge = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix} \quad (\text{A3})$$

According to the Equations (A1) and (A2), it can be deduced that:

$$\begin{aligned} \mathbf{R}_{cc'} &= (\mathbf{l} \cdot \mathbf{r}) \mathbf{I} + \frac{(\mathbf{l} \times \mathbf{r})(\mathbf{l} \times \mathbf{r})^T}{1 + \mathbf{l} \cdot \mathbf{r}} + (\mathbf{l} \times \mathbf{r})^\wedge \\ &= \begin{bmatrix} 1 - \frac{\cos^2 \varphi \sin^2 \theta}{1 + \cos \theta} & -\frac{\sin^2 \theta \cos 2\varphi}{2(1 + \cos \theta)} & \cos \varphi \sin \theta \\ -\frac{\sin^2 \theta \cos 2\varphi}{2(1 + \cos \theta)} & 1 - \frac{\sin^2 \varphi \cos^2 \theta}{1 + \cos \theta} & \sin \varphi \sin \theta \\ -\cos \varphi \sin \theta & -\sin \varphi \sin \theta & \cos \theta \end{bmatrix} \end{aligned} \quad (\text{A4})$$

### Appendix A.2. Roll Transformation

The transition coordinate system  $O_{c'}X_{c'}Y_{c'}Z_{c'}$  rotates  $\psi$  angle around  $Z$ -axis according to the right-hand rule to obtain the transition coordinate system  $O_{c''}X_{c''}Y_{c''}Z_{c''}$ . The transformation matrix between  $O_{c'}X_{c'}Y_{c'}Z_{c'}$  and  $O_{c''}X_{c''}Y_{c''}Z_{c''}$  is:

$$\mathbf{R}_{c'c''} = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A5})$$

### Appendix A.3. Translation Transformation

The transition coordinate system  $O_{c''}X_{c''}Y_{c''}Z_{c''}$  is transformed into the test image coordinate system  $0_tX_tY_tZ_t$  through a translation transformation by vector  $\mathbf{q}$ . The coordinate of vector  $\mathbf{q}$  in the test image coordinate system is defined as  $\mathbf{q}_t$ . Note that the  $4 \times 4$  relative pose matrix between  $O_{c''}X_{c''}Y_{c''}Z_{c''}$  and  $0_tX_tY_tZ_t$  is  $\mathbf{T}_{c''t}$ , which is expressed as:

$$\mathbf{T}_{c''t} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{q}_t \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (\text{A6})$$

In conclusion, the relative pose matrix  $\mathbf{T}_{ct}$ , between the camera coordinate system  $0_cX_cY_cZ_c$  and the test image coordinate system  $0_tX_tY_tZ_t$  is as follows:

$$\begin{aligned} \mathbf{T}_{ct} &= \mathbf{T}_{cc'}\mathbf{T}_{c'c''}\mathbf{T}_{c''t} \\ &= \begin{bmatrix} \mathbf{R}_{cc'} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{c'c''} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{q}_t \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_{cc'}\mathbf{R}_{c'c''} & \mathbf{R}_{cc'}\mathbf{R}_{c'c''}\mathbf{q}_t \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_{ct} & \mathbf{R}_{ct}\mathbf{q}_t \\ \mathbf{0}^T & 1 \end{bmatrix} \end{aligned} \quad (\text{A7})$$

where

$$\mathbf{R}_{ct} = \begin{bmatrix} 1 - \frac{\cos^2 \varphi \sin^2 \theta}{1 + \cos \theta} & -\frac{\sin^2 \theta \cos 2\varphi}{2(1 + \cos \theta)} & \cos \varphi \sin \theta \\ -\frac{\sin^2 \theta \cos 2\varphi}{2(1 + \cos \theta)} & 1 - \frac{\sin^2 \varphi \cos^2 \theta}{1 + \cos \theta} & \sin \varphi \sin \theta \\ -\cos \varphi \sin \theta & -\sin \varphi \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A8})$$

## Appendix B. Virtual Dataset Generation

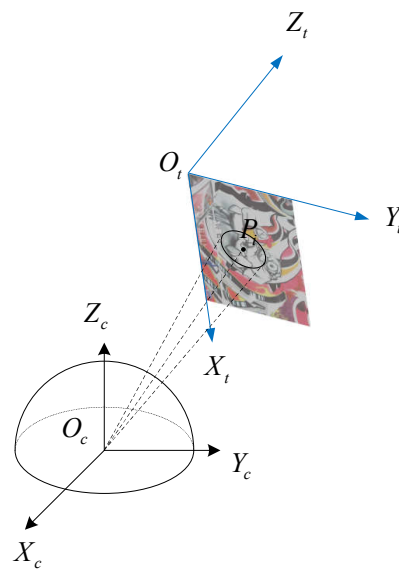
Several feature points, which are expressed as  $\{\mathbf{p}_t^i | i = 1, 2, \dots, N_p\}$ , are extracted based on FAST feature in the original Graffiti test image, where  $\mathbf{p}_t^i = [p_{tx}^i \ p_{ty}^i]^T$ ,  $N_p$  is the number of feature points, and in this experiment, the value of  $N_p$  is 30. Define that the three-dimensional point corresponding to the feature point  $\mathbf{p}_t^i$  in the test image is  $P^i$ , and the 3D coordinate of  $P^i$  in the test image coordinate system is  $\mathbf{P}_t^i$ , and  $\mathbf{P}_t^i = [p_{tx}^i \ p_{ty}^i \ 0]^T$ . Based on the original ORB centroid calculation method, the gray centroid  $\mathbf{c}_t^i$  of each feature point  $\mathbf{p}_t^i$  is calculated. Define that the corresponding 3D point of  $\mathbf{c}_t^i$  is  $C^i$ , and the 3D coordinate of  $C^i$  in the test image coordinate system is  $\mathbf{C}_t^i$ .

In order to ensure that the generated dataset can accurately test the accuracy of the algorithm to calculate the direction of feature points, the dataset generation meets the following conditions:

1. As shown in Figure A3, ensure that the line  $\overline{P^iO_c}$  is perpendicular to the test image plane  $X_tO_tY_t$ , that is  $\overline{P^iO_c} \perp X_tO_tY_t$ , so as to ensure that the circular neighborhood used to calculate the gray centroid of the feature point  $\mathbf{p}_t^i$  in the test image and the optical center  $O_c$  of the camera forms a regular cone;

2. Ensure that the length of  $\overline{P^i O_c}$  is equal to the average values of the horizontal and vertical focal length  $\frac{f_x+f_y}{2}$  of the virtual camera, so as to ensure that the circular area used for calculating the gray centroid of feature points in the original test image is approximately the same as that used to calculate the gray centroid of feature points in the fisheye image.

If these conditions are not satisfied, it will lead to the inconsistency between the calculation area of the gray centroid in the original test image and that in the fisheye image, and the calculated gray centroid will have different mathematical meanings, which will lead to the loss of the experimental verification value.



**Figure A3.** Position relationship between the camera coordinate system and the test image coordinate system.

The necessary and sufficient condition for satisfying the above rules is that the vector  $\mathbf{q}_t$  satisfies:

$$\mathbf{q}_t(\mathbf{p}_t^i) = [-p_{tx}^i \quad -p_{ty}^i \quad \frac{f_x+f_y}{2}]^T \quad (\text{A9})$$

where  $p_{tx}^i$  and  $p_{ty}^i$  are the pixel coordinates of point  $\mathbf{P}_t^i$  in the test image, and  $f_x$  and  $f_y$  is the horizontal and vertical focal length of the virtual camera.  $\varphi$  and  $\theta$  determine the projection position of point  $\mathbf{p}_t^i$  in the fisheye image.  $\psi$  determines the projection position of the gray centroid  $C^i$  in the fisheye image.

Then, the dataset is generated according to the following method:

1. Within the camera's FoV, starting from  $\theta = 10^\circ$ , taking  $10^\circ$  as the interval, the  $\theta$  angle is uniformly selected, and  $N_\theta$  values of  $\theta$  are generated.
2. From  $45^\circ$  to  $315^\circ$ ,  $\varphi$  is uniformly taken at  $90^\circ$  intervals, and the number of  $\varphi$  values generated is  $N_\varphi = 4$ .
3. When  $\varphi$  is  $45^\circ$  or  $225^\circ$ , the value of  $\psi$  is taken as  $4\theta$ . When  $\varphi$  is  $135^\circ$  or  $315^\circ$ , the value of  $\psi$  is taken as 0.

For each combination of  $\varphi$ ,  $\theta$  and  $\mathbf{p}_t^i$ , a corresponding fisheye distortion image  $I(\varphi, \theta, \mathbf{p}_t^i)$  is generated, which constitutes a virtual dataset containing  $N_\theta \times N_\varphi \times N_p$  fisheye images as follow:

$$\{I(\varphi, \theta, \mathbf{p}_t^i) | \varphi = 45^\circ, 135^\circ, 225^\circ, 315^\circ; \theta = 10^\circ, 20^\circ, \dots, 80^\circ; i = 1, 2, \dots, N_p; N_p = 30\} \quad (\text{A10})$$



The coordinate  $\mathbf{P}_c^i$  of point  $P^i$  and  $\mathbf{C}_c^i$  of point  $C^i$  in the camera coordinate system are calculated by using the coordinate transformation relationship, as shown in the following Equations:

$$\begin{aligned}\mathbf{P}_c^i &= \mathbf{T}_{ct}(\varphi, \theta, \mathbf{p}_t^i) \mathbf{P}_t^i \\ \mathbf{C}_c^i &= \mathbf{T}_{ct}(\varphi, \theta, \mathbf{p}_t^i) \mathbf{C}_t^i\end{aligned}\quad (\text{A11})$$

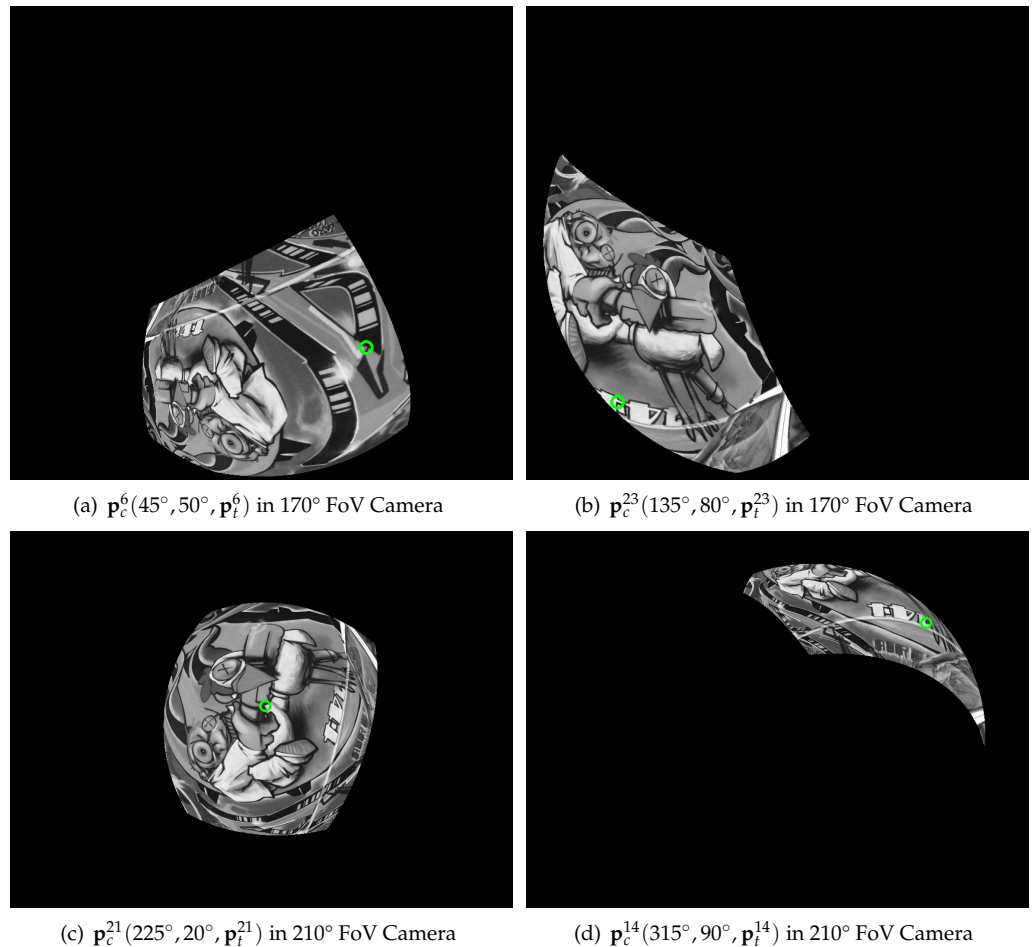
The projection position  $\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i)$  of the feature point  $\mathbf{p}_t^i$  in the fisheye image is calculated by the following formula:

$$\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i) = \Pi(\mathbf{P}_c^i) \quad (\text{A12})$$

According to the relationship shown in the following formula, the ground truth attitude matrix  $\mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i)$  corresponding to the feature point  $\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i)$  in the fisheye image is calculated by:

$$\mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i) = \begin{bmatrix} \frac{\mathbf{C}_c^i - \frac{\mathbf{C}_c^i \cdot \mathbf{C}_c^i}{\mathbf{P}_c^i \cdot \mathbf{P}_c^i} \mathbf{P}_c^i}{\left| \mathbf{C}_c^i - \frac{\mathbf{C}_c^i \cdot \mathbf{P}_c^i}{\mathbf{P}_c^i \cdot \mathbf{P}_c^i} \mathbf{P}_c^i \right|} & \frac{\mathbf{P}_c^i \times \mathbf{C}_c^i}{\left| \mathbf{P}_c^i \times \mathbf{C}_c^i \right|} & \mathbf{P}_c^i \end{bmatrix} \quad (\text{A13})$$

In this dataset, each virtual fisheye image  $I(\varphi, \theta, \mathbf{p}_t^i)$  uniquely corresponds to a feature point pixel coordinate  $\mathbf{p}_c^i(\varphi, \theta, \mathbf{p}_t^i)$  and a ground truth attitude matrix  $\mathbf{R}_{cb}^{i*}(\varphi, \theta, \mathbf{p}_t^i)$ , which constitutes a feature point test sample. Some examples of test samples are shown in Figure A4.



**Figure A4.** Some examples of virtual dataset test sample. The green circle in sample image indicates the feature point.

## References

1. Lowe, D.G. Object recognition from local scale-invariant features. *Proc. IEEE Int. Conf. Comput. Vis.* **1999**, *2*, 1150–1157.
2. Bay, H.; Tuytelaars, T.; Gool, L.V. SURF: Speeded up robust features. In *Computer Vision - ECCV 2006, Proceedings of the 9th European Conference on Computer Vision. Proceedings, Part I (Lecture Notes in Computer Science)*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 3951, pp. 404–417.
3. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary robust independent elementary features. *Lect. Notes Comput. Sci.* **2010**, *6314 LNCS*, 778–792.
4. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV 2011)*, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
5. Alcantarilla, P.F.; Bartoli, A.; Davison, A.J. KAZE features. *Lect. Notes Comput. Sci.* **2012**, *7577 LNCS*, 214–227.
6. Leutenegger, S.; Chli, M.; Siegwart, R.Y. BRISK: Binary Robust invariant scalable keypoints. In *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011*; pp. 2548–2555.
7. Ke, Y.; Sukthankar, R. PCA-SIFT: A more distinctive representation for local image descriptors. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 27 June–2 July 2004; Volume 2, pp. II506–II513.
8. Liu, L.; Peng, F.Y.; Zhao, K.; Wan, Y.P. Simplified SIFT algorithm for fast image matching. *Infrared Laser Eng.* **2008**, *37*, 181–184.
9. Alcantarilla, P.F.; Nuevo, J.; Bartoli, A. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In *Proceedings of the BMVC 2013—Electronic Proceedings of the British Machine Vision Conference 2013*, Bristol, UK, 9–13 September 2013.
10. Tian, Y.; Fan, B.; Wu, F. L2-Net: Deep learning of discriminative patch descriptor in Euclidean space. In *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, 21–26 July 2017; pp. 6128–6136.
11. Mishchuk, A.; Mishkin, D.; Radenovi, F.; Matas, J. Working hard to know your neighbor’s margins: Local descriptor learning loss. *arXiv* **2017**, arXiv:1705.10872.
12. Mishkin, D.; Radenovic, F.; Matas, J. Repeatability Is Not Enough: Learning Affine Regions via Discriminability. In *Proceedings of the ECCV, Munich, Germany, 8–14 September 2018*.
13. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM. *arXiv* **2020**, arXiv:2007.11898.
14. Urban, S.; Hinz, S. MultiCol-SLAM—a modular real-time multi-camera SLAM system *arXiv*. *arXiv* **2016**, arXiv:1610.07336.
15. Lin, Y.; Gao, F.; Qin, T.; Gao, W.; Liu, T.; Wu, W.; Yang, Z.; Shen, S. Autonomous aerial navigation using monocular visual-inertial fusion. *J. Field Robot.* **2018**, *35*, 23–51. [[CrossRef](#)]
16. Miiller, M.G.; Steidle, F.; Schuster, M.J.; Lutz, P.; Maier, M.; Stoneman, S.; Tomic, T.; Sturzl, W. Robust Visual-Inertial State Estimation with Multiple Odometries and Efficient Mapping on an MAV with Ultra-Wide FOV Stereo Vision. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 1–5 October 2018; pp. 3701–3708.
17. Yahui, W.; Shaojun, C.; Shi-Jie, L.; Yun, L.; Yangyan, G.; Tao, L.; Ming-Ming, C. CubemapSLAM: A piecewise-pinhole monocular fisheye SLAM system. In *Computer Vision-ACCV 2018, Proceedings of the 14th Asian Conference on Computer Vision. Revised Selected Papers: Lecture Notes in Computer Science (LNCS 11366)*; Jawahar, C., Li, H., Mori, G., Schindler, K., Eds.; Springer: Cham, Switzerland, 2019; Volume 11366, pp. 34–49.
18. Scaramuzza, D.; Siegwart, R. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Trans. Robot.* **2008**, *24*, 1015–1026. [[CrossRef](#)]
19. Tardif, J.P.; Pavlidis, Y.; Daniilidis, K. Monocular visual odometry in urban environments using an omnidirectional camera. In *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Nice, France, 22–26 September 2008*; pp. 2531–2538.
20. Rituerto, A.; Puig, L.; Guerrero, J.J. Visual SLAM with an Omnidirectional Camera. In *Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010*; pp. 348–351. [[CrossRef](#)]
21. Arican, Z.; Frossard, P. OmniSIFT: Scale invariant features in omnidirectional images. In *Proceedings of the International Conference on Image Processing, ICIP, Hong Kong, China, 26–29 September 2010*; pp. 3505–3508.
22. Lourenco, M.; Barreto, J.P.; Vasconcelos, F. SRD-SIFT: Keypoint detection and matching in images with radial distortion. *IEEE Trans. Robot.* **2012**, *28*, 752–760. [[CrossRef](#)]
23. Cruz Mota, J.; Bogdanova, I.; Paquier, B.; Bierlaire, M.; Thiran, J.P. Scale invariant feature transform on the sphere: Theory and applications. *Int. J. Comput. Vis.* **2012**, *98*, 217–241. [[CrossRef](#)]
24. Hansen, P.; Corke, P.; Boles, W. Wide - angle visual feature matching for outdoor localization. *Int. J. Robot. Res.* **2010**, *29*, 267–297. [[CrossRef](#)]
25. Zhao, Q.; Feng, W.; Wan, L.; Zhang, J. SPHORB: A Fast and Robust Binary Feature on the Sphere. *Int. J. Comput. Vis.* **2015**, *113*, 143–159. [[CrossRef](#)]
26. Urban, S.; Weinmann, M.; Hinz, S. mdBRIEF—a fast online-adaptable, distorted binary descriptor for real-time applications using calibrated wide-angle or fisheye cameras. *Comput. Vis. Image Underst.* **2017**, *162*, 71–86. [[CrossRef](#)]
27. Pourian, N.; Nestares, O. An End to End Framework to High Performance Geometry-Aware Multi-Scale Keypoint Detection and Matching in Fisheye Imag. In *Proceedings of the International Conference on Image Processing, ICIP, Taipei, Taiwan, 22–25 September 2019*; pp. 1302–1306.

- 
28. Kannala, J.; Brandt, S.S. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 1335–1340. [[CrossRef](#)] [[PubMed](#)]
  29. Viswanathan, D.G. Features from accelerated segment test (fast). In Proceedings of the 10th workshop on Image Analysis for Multimedia Interactive Services, London, UK, 6 May 2009; pp. 6–8.
  30. Mikolajczyk, K.; Schmid, C. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1615–1630. [[CrossRef](#)] [[PubMed](#)]