Benefit of Large Field-of-View Cameras for Visual Odometry

Zichao Zhang, Henri Rebecq, Christian Forster, Davide Scaramuzza

***Abstract*— The transition of visual-odometry technology from research demonstrators to commercial applications naturally raises the question: "*what is the optimal camera for vision-based motion estimation?*" This question is crucial as the choice of camera has a tremendous impact on the robustness and accuracy of the employed visual odometry algorithm. While many properties of a camera (e.g. resolution, frame-rate, global-shutter/rolling-shutter) could be considered, in this work we focus on evaluating the impact of the camera field-of-view (FoV) and optics (i.e., fisheye or catadioptric) on the quality of the motion estimate. Since the motion-estimation performance depends highly on the geometry of the scene and the motion of the camera, we analyze two common operational environments in mobile robotics: an urban environment and an indoor scene. To confirm the theoretical observations, we implement a state-of-the-art VO pipeline that works with large FoV fisheye and catadioptric cameras. We evaluate the proposed VO pipeline in both synthetic and real experiments. The experiments point out that it is advantageous to use a large FoV camera (e.g., fisheye or catadioptric) for indoor scenes and a smaller FoV for urban canyon environments.**

Supplementary Material

A video showing our omnidirectional visual odometry pipeline performing on real and synthetic data is available at the website: `http://rpg.ifi.uzh.ch/fov.html`.

I. Introduction

Estimating the six degrees-of-freedom motion of a camera simply from its stream of images has been an active field of research for several decades [1], [2], [3]. Today, state-of-the-art algorithms run in real-time on smartphone processors and achieve the accuracy and robustness that is required to enable various interesting applications. However, the remaining challenge to enable commercial applications in risky fields such as drone delivery or autonomous driving is *robustness*, especially during fast motions, illumination changes, and in environments with difficult texture. All three nuisances increase the difficulty to track visual cues, which is fundamental to enable vision-based motion estimation.

Our work is motivated by the question of whether the robustness of existing visual odometry (VO) algorithms can be significantly improved by selecting the best camera for the task at hand. In order to minimize the design space, we limit ourselves to the selection of the optimal optics. We are particularly interested in the performance of omnidirectional cameras, which are fisheye and catadoptric cameras characterized by a large field of view (FoV). In theory, a larger FoV allows tracking visual landmarks over longer periods, which should increase the precision of pose estimation as more measurements are available and, at the same time, increase robustness since the visual overlap between subsequent images is larger. However, increasing the FoV while fixing the resolution means that the angular resolution of a pixel is reduced, hence, lowering the measurement accuracy of a single camera pixel.

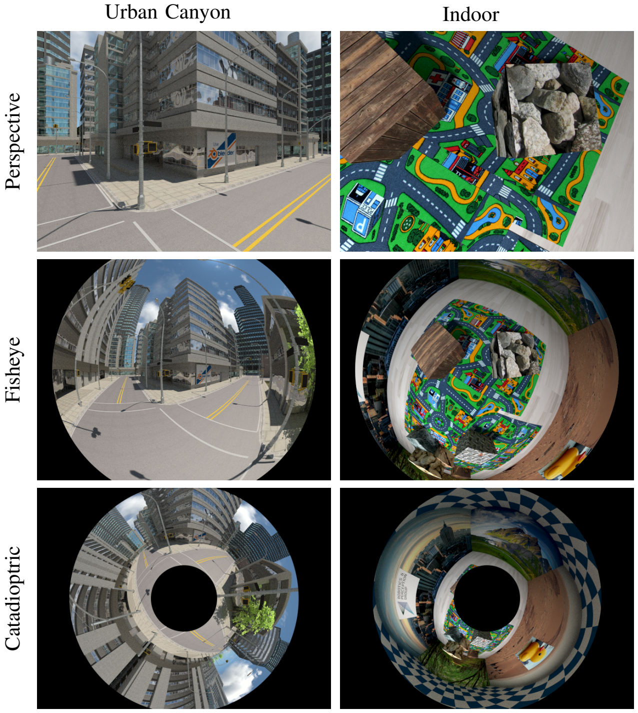


Fig. 1: Images from our synthetic datasets, showing different FoV cameras.

The contribution of this work is threefold: after discussion of related work in Section I-A, we present in Section II simulation experiments that show the impact of the FoV of a camera on the accuracy and robustness of a canonical VO pipeline. The analysis encompasses standard steps of a visual-odometry pipeline. After studying the theoretical advantages of large FoV cameras and to facilitate an analysis on real images, we describe in Section III challenges and solutions to enable a state-of-the-art VO pipeline (in our case *SVO* [4]) to operate with such images. Therefore, we provide a detailed study of six error metrics on the pose estimation accuracy. Our analysis helps to select the proper error metrics as a function of the camera FoV. Finally, in Section IV, we evaluate the performance of the proposed omnidirectional SVO algorithm in synthetic as well as real experiments for various camera optics. Since the impact of the camera FoV is a function of the application scenario, we perform the experiments in different environments that

The authors are with the Robotics and Perception Group, University of Zurich, Switzerland `http://rpg.ifi.uzh.ch`. This research was funded by the China Scholarship Council.

reflect typical applications of VO (e.g., automotive, drones, gaming). As a further contribution, we publicly release all our synthetic and real datasets that we recorded with different FoV cameras[1].

A. Related Work

The type of camera used for vision-based navigation methods has a significant impact on the accuracy and robustness of the motion estimation process. A comparison of the performance of a catadioptric and a perspective camera in a visual SLAM system was presented in [5]. A catadioptric camera has a shaped mirror mounted in the front that allows it to capture the full 360 degree view. Experimental results showed that the catadioptric camera outperforms the perspective camera in terms of motion estimation accuracy. However, the catadioptric camera that was used for the experiments had a higher pixel resolution than the perspective camera. Thereby, the lower angular resolution of the larger FoV catadioptric camera was compensated, which provided an unfair advantage to the catadioptric camera. Nevertheless, the comparison presented in [6] experimentally confirmed that a larger FoV camera has a higher motion estimation accuracy than a smaller FoV perspective camera even in the case of a fixed pixel resolution. Unfortunately, the experiments were limited to synthetic data and an indoor environment. In our experiments we confirm these results in an indoor scenario, but we show, both on synthetic and real data, that large FoV cameras perform worse than standard perspective cameras in outdoor environments.

Most VO algorithms for omnidirectional cameras [7], [8], [9], [10] rely on robust feature descriptors (e.g., SIFT [11]) to establish feature correspondence. To cope with the significant distortion of large FoV images, special descriptors were developed that model the distortion effects to improve feature matching [12], [13], [14], [15]. Other works, such as [16] and [17], used Lucas-Kanade feature tracking [18] to estimate the motion of landmark observations between frames of omnidirectional images.

In this work, we develop a VO pipeline for omnidirectional cameras based on the state-of-the-art *Semi-direct Visual Odometry* (SVO) algorithm [4]. SVO is a very fast odometry algorithm because it does not extract salient features in every frame. Instead, it uses a direct method to estimate the camera motion by mimizing the photometric error of corresponding pixels in subsequent views, similar to LSD [19] and DTAM [20]. However, in contrast to LSD and DTAM, the so called *sparse image alignment* step in SVO works only with sparse pixels and, thus, the convergence radius of the alignment is small and can only be applied on a frame-to-frame basis. Therefore, given the frame-to-frame pixel correspondence, which is found by means of sparse image alignment, the SVO pipeline uses a classic feature-based nonlinear refinement step to minimize the drift. In Section III we describe the required modifications to the standard SVO[2] to enable motion estimation with cameras that have a FoV larger than 120 degrees.

In the next section, we will study the impact of a large FoV on the performance of VO.

II. Optimal Field-of-View Studies for Canonical Visual Odometry Pipeline

In this section, we study the impact of the camera FoV on a canonical VO pipeline by means of Monte Carlo simulations. First, we present a study of the influence of the FoV on the accuracy of three standard components of a VO pipeline: feature correspondence, pose optimization and combined map-pose estimation. By pose optimization we denote the nonlinear refinement of the camera pose, which minimizes the reprojection error of known 3D landmarks. Note that this step is typically applied in an odometry pipeline after finding a solution to the perspective-n-point (PnP) problem. The third experiment implements a canonical VO pipeline combining both depth estimation and pose optimization.

As we will see, the optimal FoV depends greatly on the structure of the environment. Therefore, we perform the study in two different simulated scenes: in the first scene the camera moves in an *urban canyon* that simulates an automotive setting, while, in the second environment, the camera moves in a *confined room* that simulates common indoor scenarios. We evaluate the second scene both with a forward- and downward-looking camera.

A. Experiment 1: Feature Correspondence

The foundation of all geometric vision problems is *feature correspondence*. Hence, the accuracy of 3D landmark measurements (i.e., keypoints) in the images directly affects the accuracy of the motion estimate. Therefore, our first experiment evaluates the accuracy of feature correspondence for three different cameras with a constant image resolution. The experiment is based on synthetic scenes rendered for different FoV cameras using Blender (Fig. 1). Given a keypoint in a reference image, we search for the corresponding keypoint in a subsequent image of the same camera trajectory by means of Lucas-Kanade feature alignment [18]. The groundtruth of the keypoint alignment is calculated by first backprojecting the keypoint from the reference image to the 3D model of the scene to get the 3D landmark and then projecting the landmark to the subsequent frame.

Figure 2 shows the alignment error as a function of the distance to the reference view. We observe that the accuracy of feature correspondence decreases as we select a frame in the camera trajectory that is farther from the reference frame. Also, the accuracy is slightly reduced when the cameras with larger FoVs are used. The reason for this is that for larger FoV cameras, the image patches used in the alignment suffer from more severe distortions between the reference frame and the selected frame. Given these considerations, in the following experiments we corrupt all feature correspondences with zero-mean additive white Gaussian noise with

[1] Available at http://rpg.ifi.uzh.ch/fov.html

[2] Available at http://github.com/uzh-rpg/rpg_svo

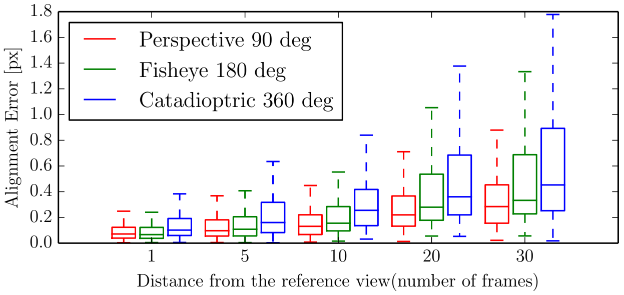


Fig. 2: *Experiment 1*: Keypoint alignment accuracy for different optics as a function of the distance from the reference frame.

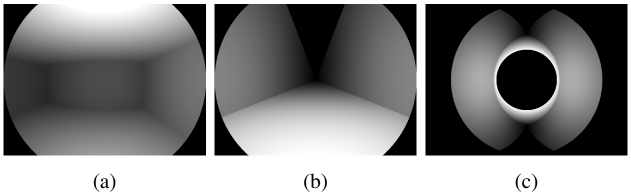
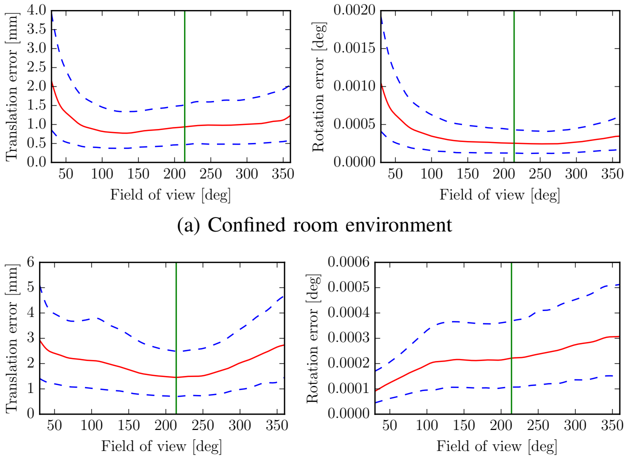


Fig. 3: Rendered images showing what the camera sees in different setups: front-looking camera in box environment, front-looking camera in canyon environment, up-looking catadioptric camera in canyon environment. Note that the texture is not given because the groundtruth depth is available.



(b) Urban canyon environment

Fig. 4: *Experiment 2*: Pose estimation accuracy with respect to FoV for two synthetic scenes. Solid line is the median; dashed lines bound the confidence interval.

$\sigma = 0.25$ pixels, which reflects the average uncertainty of our measurements.

B. Experiment 2: Pose Optimization

The pose optimization step refines the pose $\mathtt{T}_{\mathrm{CW}} \in \mathrm{SE}(3)$ of the camera C with respect to a world frame W by minimizing the reprojection error of the visible landmarks. Hence, we are solving the following nonlinear least-squares problem:

$$\mathtt{T}_{\mathrm{CW}} = \arg\min_{\mathtt{T}} \frac{1}{2} \sum_{i=1}^{N} \| \mathbf{r}(\tilde{\mathbf{u}}_i, \pi(\mathtt{T}\ {}_{\mathrm{W}}\mathbf{p}_i)) \|^2, \qquad (1)$$

where ${}_{\mathrm{W}}\mathbf{p}_i \in \mathbb{R}^3$ are the landmark positions expressed in the world frame. The metric we use for the reprojection residual $\mathbf{r}(\tilde{\mathbf{u}}_i, \hat{\mathbf{u}}_i)$ between the measured feature position $\tilde{\mathbf{u}}_i \in \mathbb{R}^2$ and the predicted feature position $\hat{\mathbf{u}}_i = \pi(\mathtt{T}\ {}_{\mathrm{W}}\mathbf{p}_i) \in \mathbb{R}^2$ is discussed in more detail in Section III-B. By $\pi : \mathbb{R}^3 \to \mathbb{R}^2 : \mathbf{u} = \pi(\mathbf{p})$ we denote the camera projection function.

In this section, we assume that a perfectly known 3D map of the environment is available, whereas in the next section the map is computed using triangulation.

For this experiment, we simulate cameras with varying FoVs using the equidistant fisheye model [21]. The image resolution is fixed, thus the angular resolution decreases as the FoV increases. A forward-looking camera is placed in the center of the scene (Fig. 3). For each feature in the image plane, the corresponding visible 3D point is found using raytracing on the synthetic scenes. We sample 150 features uniformly in the image plane and compute their corresponding 3D landmarks. Features are corrupted as described in Section II-A. With these inputs (2D-3D correspondences), we solve the absolute pose estimation problem. The experiment is repeated 1000 times for each FoV.

Fig. 4 shows the pose estimation accuracy as a function of the FoV, for the confined room and canyon scenes. It can be observed, that larger FoV cameras perform better in the room scene, despite the loss of angular resolution. Indeed, increasing the FoV yields more evenly distributed landmarks in space (as a larger FoV allows to capture points with a greater angular distance to the optical axis), which stabilizes the pose optimizer (this was also reported in [6]). By contrast, in Fig. 4b, the translation error reaches a minimum for a FoV of about 215 degrees. This can be interpreted as the result of two competing effects. On the one hand a larger FoV provides a better conditioning for the PnP problem, which raises the pose estimation accuracy. On the other hand, as the FoV grows, the angular resolution decreases (since the image resolution is fixed), leading to larger angular errors on the landmark measurements, thus degrading the pose estimation accuracy. As shown in Fig. 4b, for the canyon scene, the first effect prevails for small and moderate FoVs while the second eventually becomes predominent for very large FoVs.

Note that this experiment was conducted using a synthetic camera, allowing for arbitrarily large FoVs. While, in reality, fisheye lenses typically reach a maximum FoV of approximately 215° (e.g., the KodakSP360 camera), this experiment still provides some valuable insight on the trade-off involved when selecting an optics for a given sensor. The vertical line in Fig. 4 marks the frontier between existing and purely synthetic cameras.

C. Experiment 3: Canonical Visual Odometry Pipeline

This section assumes no prior knowledge of the map, therefore in the following experiment we simulate a full VO pipeline: from noisy observations we triangulate 3D landmarks that are used to estimate the camera pose of subsequent images (see Fig. 5). This is a standard approach for incremental camera motion estimation [10].

We simulate a camera trajectory (Fig. 6) in the desired environment and select a reference *keyframe* (red in Fig. 5) among the trajectory frames. As in the previous experiment,

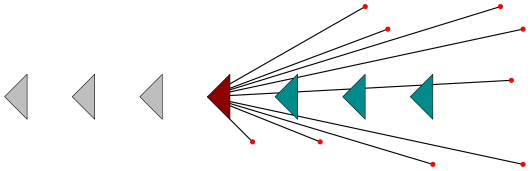
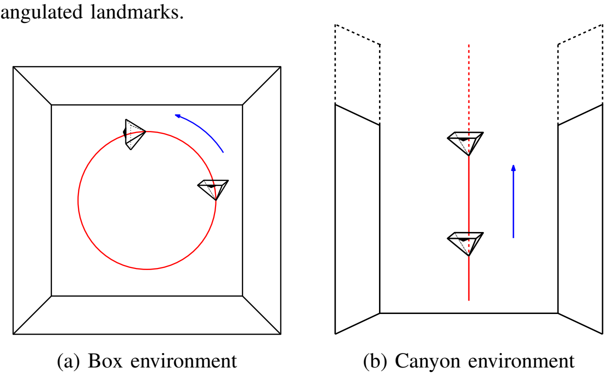


Fig. 5: *Experiment 3*: Camera moving along the trajectory, keyframes and triangulated landmarks.



(a) Box environment

(b) Canyon environment

Fig. 6: *Experiment 3*: Top views of the different setups. For the box scene, the experiment is conducted with both downward-looking and forward-looking camera but only the latter is shown in this figure.

we sample features uniformly in the reference keyframe image plane. Corresponding landmarks (red dots) are triangulated using a set of previous frames (shown in grey), projected and corrupted in the image plane as before. Then, the poses of the following frames (green) are estimated based on the triangulated landmarks. This experiment is conducted for various camera FoVs on both synthetic scenes, with 1000 runs for each configuration. Additionally, in two cases, an up-looking catadioptric camera with a horizontal FoV of 360° and vertical FoV from -50° to +50° above the horizon is simulated.

The results of our experiment are shown in Fig. 7. The pose estimation accuracy is evaluated as a function of the distance to the keyframe. This provides a measure of robustness and drift: Robustness is increased if we can move farther away from the last keyframe without loosing much pose accuracy, whereas drift is reduced if we can track features over longer time intervals.

The main conclusion from these experiments is that, for visual localization, large FoV cameras should be preferred in confined environments (e.g., indoor flight for a drone), whereas smaller FoV cameras will perform better for forward-looking cameras in canyon-like environments (typically a camera mounted on a car in the city). Specifically, the analysis of the plots in Fig. 7 follows.

a) Room environment: Regardless of the camera orientation, the motion estimation accuracy grows with the FoV (Figs. 7a and 7b). The superiority of wide angle optics in this setup stems from two different beneficial effects: first, the better angular distribution of features, as demonstrated in Section II-B; and second, the ability of large FoV cameras to track features longer greatly increases the robustness of visual localization in this environment (see Fig. 7b: almost all features remain visible as the down-looking camera moves). Interestingly, the catadioptric camera performs slightly worse than the large FoV fisheye cameras. This is consistent with the results from the previous section: the localization accuracy stops increasing when the FoV reaches a threshold of around 210 degrees, and the catadioptric camera's self-occlusion zone furthermore reduces the available image area compared to the fisheye cameras.

b) Front-looking camera in canyon environment: This experiment (Fig. 7c) shows that a smaller FoV should be preferred in an urban canyon scenario. The reason why large FoV optics perform worse in this setup is twofold. Firstly, because the depth range of the scene is much higher than the room scene. Whereas the triangulation error introduced by the loss of angular resolution remains small when the depth range of the landmarks is limited, it eventually becomes predominant when the depth range is very high (in the canyon environment, the farthest point is 250m away from the camera). Secondly, because of the uniform sampling of the features in the image plane, the landmarks corresponding to the features extracted in the reference frame tend to be farther away for smaller FoV cameras, thus having a slower apparent motion with respect to the camera. These features can therefore be tracked more reliably (because of the reduced optical flow between two successive frames), and longer. Our experiment confirmed this somewhat surprising fact (third column of Fig. 7c): the camera with the smallest FoV observes features longer on average.

III. Implementation of a Semi-Direct Omnidirectional Visual Odometry

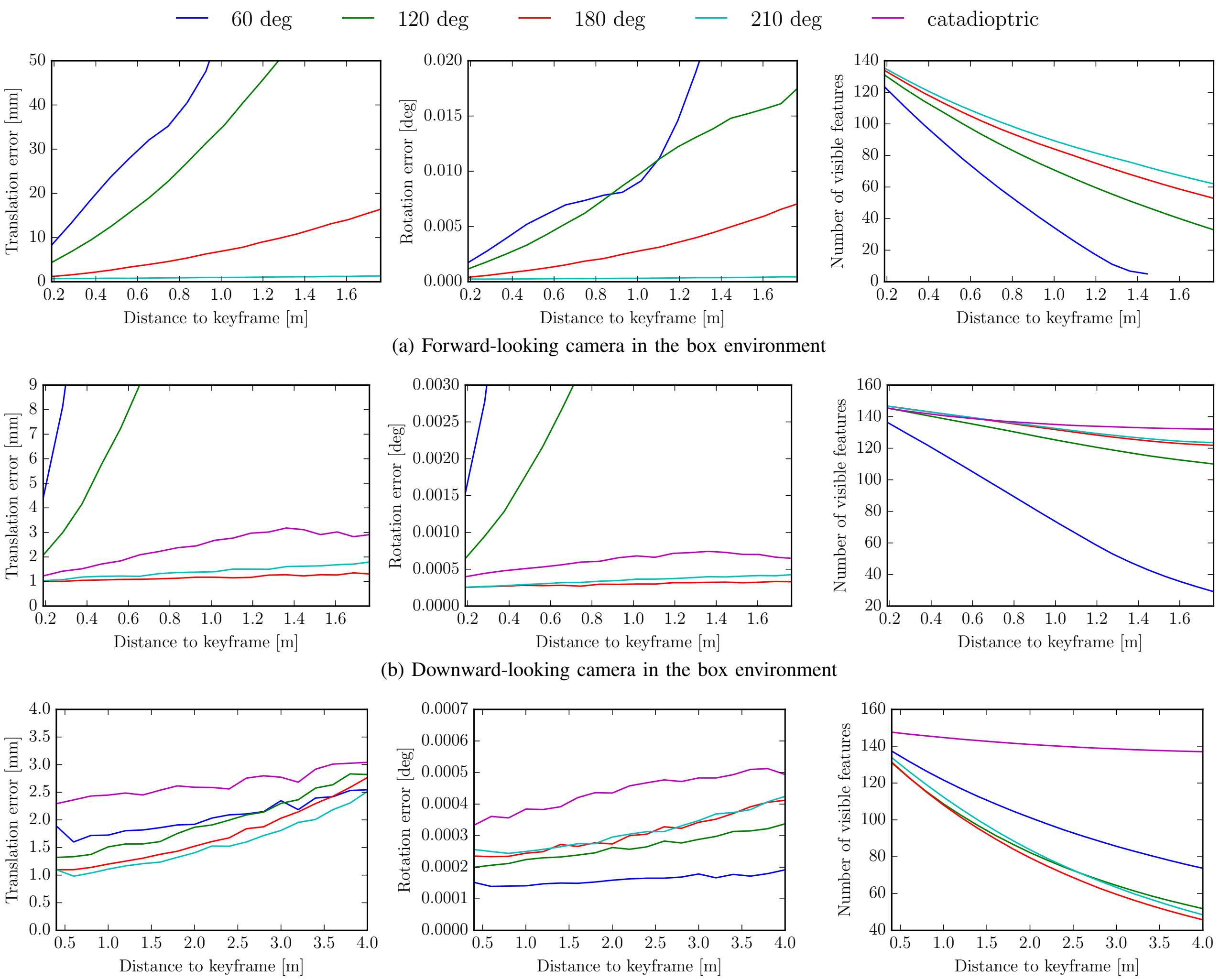
In this section, we describe the challenges and, accordingly, our solutions, to enable a state-of-the-art VO pipeline to work with wide field-of-view cameras. In particular, we develop a unified VO system that works with fisheye as well as catadioptric cameras.

We base our developments on the state-of-the-art SVO [4] pipeline. The standard SVO algorithm does not scale to large FoV cameras, which required us to perform three main modifications: (1) implementation of polynomial and equidistant camera models that adequately model large FoV cameras; (2) use of reprojection-error metrics based on bearing vectors in the pose optimization (bundle adjustment) step; (3) sampling of the curved epipolar line based on the unit sphere for better correspondence search and triangulation.

In the following, we discuss the implementation of these modifications in more detail.

A. Omnidirectional Camera Model

The omnidirectional camera model from [22] is used in our work. In this model, a Taylor series expansion is used to describe the image projection function. We choose this camera model largely due to its advantage of being able to describe catadioptric and fisheye cameras within one unified framework compared to other omnidirectional models such as the unified projection model [23] and the equidistant model [21].



(a) Forward-looking camera in the box environment

(b) Downward-looking camera in the box environment

(c) Forward-looking camera in the canyon environment

Fig. 7: *Experiment 3*: Pose error and number of visible features for different FoVs in the canonical VO pipeline.

B. Error Metrics for Pose Optimization

The SVO algorithm finds 2D-3D landmark correspondence using direct methods, specifically *sparse image alignment* and *feature alignment* [4]. In the subsequent pose optimization step, the six degree of freedom (DoF) pose of a frame is refined by minimizing the reprojection error. This problem is formalized in (1) and can be solved by standard least squares optimization techniques such as the Gauss-Newton method.

In a standard implementation, one would minimize the image error (see Fig. 8):

$$\mathbf{r}_u = \tilde{\mathbf{u}} - \pi(\mathbf{p}), \tag{2}$$

where $\mathbf{p} = [p_x, p_y, p_z]^\top$ is the 3D landmark (in the camera frame). However, this requires to compute the projection function and its Jacobian at each iteration, which can be expensive when complicated camera models are used. Therefore, SVO minimized the reprojection error on the *unit plane*:

$$\mathbf{r}_m = \tilde{\mathbf{m}} - \left[\frac{p_x}{p_z}, \frac{p_y}{p_z}\right]^\top, \tag{3}$$

where $\tilde{\mathbf{m}}$ is the corresponding position of observation $\tilde{\mathbf{u}}$ on the unit plane. Unfortunately, this approach does not scale when the FoV is large as p_z approaches zero for landmarks observed at the border of the image. Hence, implementations of omnidirectional vision systems such as [24], [25] use the angular error $\Delta\theta$ between the unit bearing vectors $\tilde{\mathbf{f}}$ and $\mathbf{f}$ corresponding to $\tilde{\mathbf{u}}$ and $\mathbf{p}$, respectively:

$$\mathbf{r}_{a1} = 1 - \tilde{\mathbf{f}}^\top \mathbf{f} \quad \Longrightarrow \quad \|\mathbf{r}_{a1}\|^2 = 4\sin^4(\Delta\theta/2), \tag{4}$$

$$\mathbf{r}_{a2} = \arccos(\tilde{\mathbf{f}}^\top \mathbf{f}) \quad \Longrightarrow \quad \|\mathbf{r}_{a2}\|^2 = (\Delta\theta)^2. \tag{5}$$

Instead, the difference between the bearing vectors gives:

$$\mathbf{r}_f = \tilde{\mathbf{f}} - \mathbf{f} \quad \Longrightarrow \|\mathbf{r}_f\|^2 = 4\sin^2(\Delta\theta/2). \tag{6}$$

The authors of [26] studied different error metrics for the omnidirectional SfM problem and showed experimentally that the following tangential error was the best error metric for the pose estimation problem:

$$\mathbf{r}_t = \sqrt{\frac{2}{1 + \tilde{\mathbf{f}}^\top \mathbf{f}}}(\tilde{\mathbf{f}} - \mathbf{f}) \implies \|\mathbf{r}_t\|^2 = 4\tan^2(\Delta\theta/2). \tag{7}$$

To answer the question of which error metric to use, the same Monte Carlo experiment as in Section III-B is performed using different error metrics. The average position

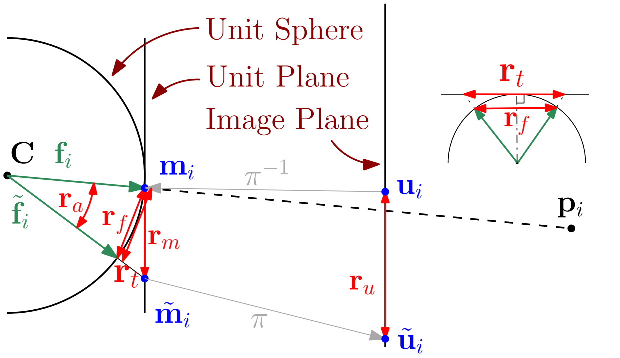


Fig. 8: Different error metrics that we evaluated for pose optimization. The landmark $\mathbf{p}_i \in \mathbb{R}^3$ is measured at pixel location $\tilde{\mathbf{u}}_i$. After applying the inverse camera projection $\tilde{\mathbf{f}}_i = \pi^{-1}(\tilde{\mathbf{u}}_i)$, which also models the distortion, we find the corresponding bearing vector $\tilde{\mathbf{f}}_i$ and unit plane coordinates $\tilde{\mathbf{m}}_i$. Given an estimate of the pose of the camera center $\mathbf{C}$, we can predict the feature position $\mathbf{u}_i = \pi(\mathbf{p}_i)$ or use intermediate results (before applying the camera distortion) to find the predicted bearing vector $\mathbf{f}_i$ or unit plane coordinates $\mathbf{m}_i$. We evaluate the efficiency and accuracy of various residual metrics $\{\mathbf{r}_{a1}, \mathbf{r}_{a2}, \mathbf{r}_t, \mathbf{r}_f, \mathbf{r}_m, \mathbf{r}_u\}$.

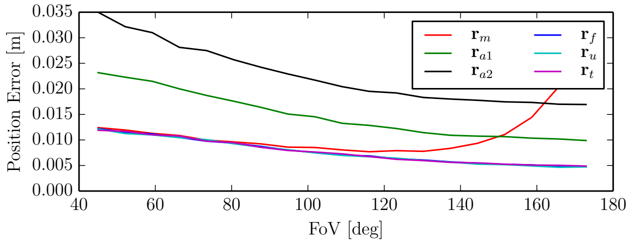


Fig. 9: Pose optimization errors of the error metrics in Fig. 8 under different FoVs. Only the position errors are given here for briefness, since the rotation and reprojection errors show a similar trend.

errors after the optimization are shown in Fig. 9. It can be observed that the image error $\mathbf{r}_u$, the tangential error $\mathbf{r}_t$ and the bearing vector difference error $\mathbf{r}_f$ have comparable performances for all the FoVs. In comparison, the unit plane error $\mathbf{r}_m$ results in equal accuracy for small FoVs, but exhibits large errors for large FoVs. When using the angular error metrics $\mathbf{r}_{a1}$ and $\mathbf{r}_{a2}$, the pose estimations oscillate around the true values instead of converging after 4-6 iterations as the other error metrics.

The time cost for each error metric is summarized in Table I. The angular error $\mathbf{r}_{a1}$ and $\mathbf{r}_{a2}$, which are not listed in the table, have a much worse time performance because of the convergence problem.

TABLE I: Average Convergence Time

	$\mathbf{r}_u$	$\mathbf{r}_m$	$\mathbf{r}_f$	$\mathbf{r}_t$
Time(ms)	0.4	0.2-0.25*	0.28	0.31

* increases with the field of view

Therefore, it can be concluded that for pose optimization, the unit plane error $\mathbf{r}_m$ should be used for small FoVs (e.g. perspective cameras with less than 100° FoVs) due to its efficiency and for large FoVs, the bearing vector difference error $\mathbf{r}_f$ should be used. In the experiments of this work, the bearing vector difference error $\mathbf{r}_f$ is used for omnidirectional cameras and the unit plane error $\mathbf{r}_m$ for perspective cameras.

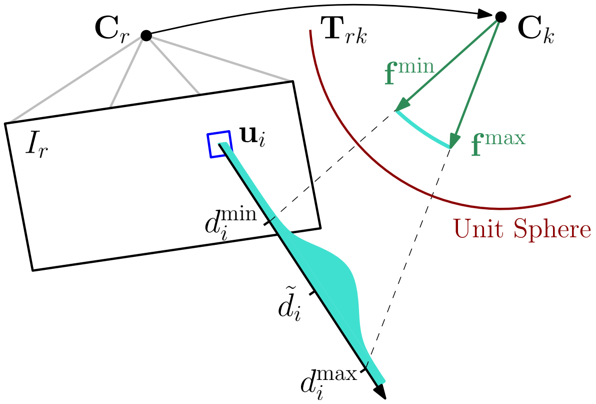


Fig. 10: Epipolar search on unit sphere for depth filter update.

C. Feature Correspondence along Curved Epipolar Lines

SVO triangulates new landmarks from known camera poses by means of a *depth filter* [4]: In a selected reference image $\mathtt{I}_r$ salient corners are selected for which the depth is estimated using measurements from older and newer frames $\mathtt{I}_k$. A measurement is obtained by sampling the epipolar line in a neighbouring image $\mathtt{I}_k$ pixel by pixel and computing the correlation of an 8×8 pixel patch with the reference patch in $\mathtt{I}_r$. The pixel on the epipolar line with highest correlation is used to update the depth of the reference pixel through triangulation (see Fig. 10).

For omnidirectional cameras, the epipolar line in $\mathtt{I}_k$ is not straight but forms a curve. To sample pixels on the curved epipolar line, we compute the bearing vectors $\{\mathbf{f}^{\min}, \mathbf{f}^{\max}\}$ that correspond to the confidence interval of the current depth estimate $d \pm 2\sigma_d = \{d^{\min}, d^{\max}\}$ in the reference image. Subsequently, we rotate a bearing vector $\mathbf{f}'$ in small angular steps from $\mathbf{f}^{\min}$ to $\mathbf{f}^{\max}$ around the axis $\mathbf{f}^{\min} \times \mathbf{f}^{\max}$ and project it on the image $\mathbf{u}' = \pi(\mathbf{f}')$, which results in a pixel location $\mathbf{u}'$ that lies on the curved epipolar line.

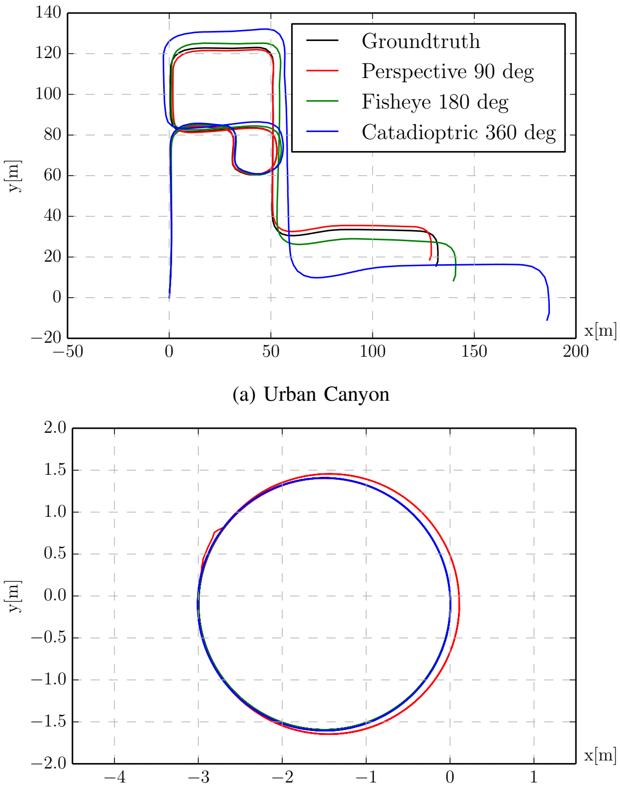
IV. Experiments

The modified SVO algorithm described in the previous section allows us to verify our FoV studies in Section II on real and synthetic images. In the following, we first discuss the synthetic experiments and subsequently the real experiments performed with a micro aerial vehicle (MAV) and an automobile.

A. Synthetic Datasets

To generate photorealistic synthetic images, we used the Cycles raytracing engine[3] implemented in Blender. In addition to the already built-in perspective and equidistant fisheye camera models, we implemented a catadioptric camera model based on [22], which we release as an open-source patch for Blender[4].

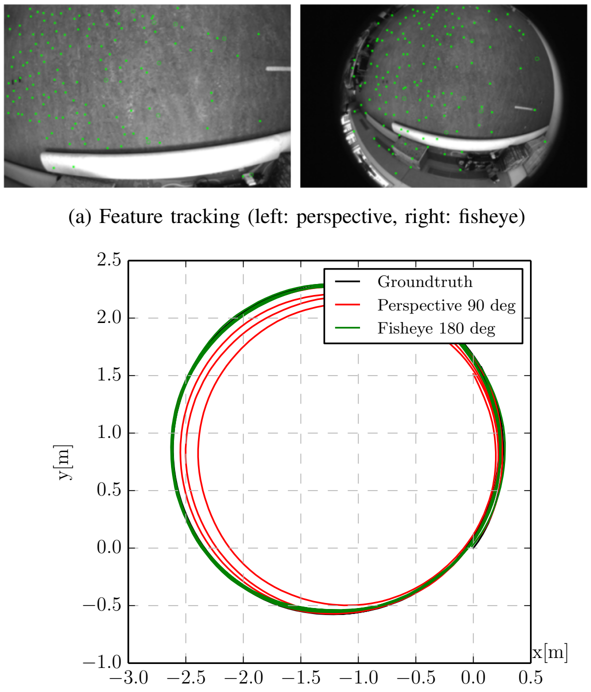
[3] http://www.blender.org/manual/render/cycles/
[4] https://github.com/uzh-rpg/rpg_blender_omni_camera



(a) Urban Canyon

(b) Indoor

Fig. 11: *Synthetic Datasets*: Top views of the estimated trajectories.



(a) Feature tracking (left: perspective, right: fisheye)

(b) Trajectory top view

Fig. 12: *Real Datasets*: Results on the *Flyroom* sequence.

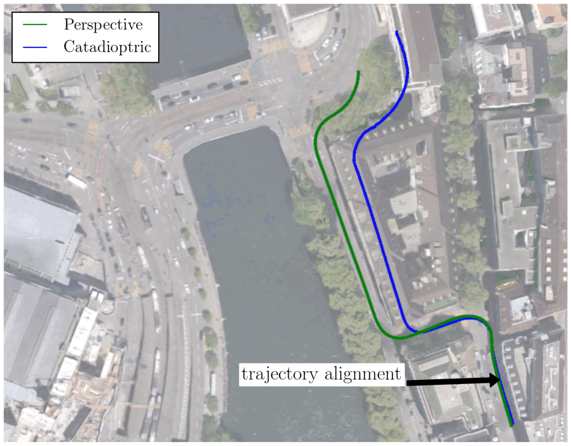


Fig. 13: *Real Datasets*: Results on the *Zurich* sequence. The first straight segment of each estimated trajectory is aligned with the corresponding part of the streets that the car drove along.

We first ran our algorithm on two synthetic datasets: *Urban Canyon* and *Indoor* (Fig. 1). The *Urban Canyon* dataset simulates a forward-looking camera mounted on a car driving in a city environment and the *Indoor* dataset contains views from a downward-looking camera moving along a circle in an indoor environment. We rendered these two datasets with three different camera models respectively: perspective (90° FoV), fisheye (180° FoV) and catadioptric (360° FoV). Note that for the catadioptric camera, the same trajectories were used for the rendering but the camera was set up to be upward-looking (facing the mirror).

The top view of the trajectories estimated is shown in Fig. 11. It can be observed that the perspective camera exhibits the smallest drift in the *Urban Canyon* dataset, followed by the fisheye camera and the catadioptric camera. However, in the *Indoor* dataset, while the trajectories estimated by the omnidirectional cameras are almost identical to the groundtruth, the perspective camera exhibits significant drift.

B. Real Datasets

To further verify our FoV studies with real world scenarios, we first recorded a *Flyroom* dataset with a downward-looking camera mounted on a MAV. The camera was 1 m above the ground and moved along a circle of about 1.5 m radius at a speed of 1.3 m/s. The datasets were recorded with a perspective camera (90° FoV) and a fisheye one (180° FoV), respectively. The groundtruth was acquired via a motion capture system. Fig. 12 shows the performance comparison between the two cameras. It can be observed from Fig. 12b that the trajectory estimated by the fisheye camera follows the circle precisely, while the trajectory estimated by the perspective one drifts away as it repeats the circle. It can be seen from Fig. 12a that while the perspective camera can only track features that are very close, the fisheye one can keep track of features from a much larger area.

We also ran our algorithm on the *Zurich* dataset from [27]. The *Zurich* dataset contains two sequences: a forward-looking perspective camera (45° FoV) and an upward-looking catadioptric camera (360° FoV).The two sequences were recorded on the same car simultaneously while the car drove through Zurich downtown. Since no groundtruth is available for this dataset, the estimated trajectories were aligned with a satellite map for evaluation. As is shown in Fig. 13, the trajectory estimated with the perspective camera is more consistent with the streets on the map.

C. Discussion

The results from the above experiments are consistent with our simulations and analysis presented in Section II.

- For indoor scenarios, such as the *Indoor* and *Flyroom* datasets, large FoV omnidirectional cameras outperform the perspective ones. The reason for this is twofold: first, features are more evenly distributed in space, which stabilizes the pose estimation, and, second, the camera can track features for a longer time.
- For outdoor environments such as the *Urban Canyon* and *Zurich* datasets, the trajectories can be estimated more accurately using perspective cameras, mainly because the loss of angular resolution for higher FoVs is drastically amplified by the higher depth range.

V. Conclusions

It is well known that VO can benefit from large FoVs. Indeed, a larger FoV theoretically allows for tracking visual landmarks over longer periods, which should increase the precision of pose estimation (since more measurements are available) and increase robustness since the visual overlap between successive images is larger. However, at the same time, increasing the FoV while fixing the resolution decreases the angular resolution of the image, thus, lowering the measurement accuracy of a single camera pixel.

In this work, we showed that for a constant image resolution, the best choice of FoV and optics is not as straightforward as it seems. We first performed extensive simulations to study the impact of different FoVs on the standard VO modules as well as the complete pipeline, which point out that large FoV cameras (e.g., omnidirectional cameras) are preferable in indoor environments, while smaller FoV cameras perform better in urban canyon scenarios. We also performed experiments using both synthetic and real world datasets and these are in accordance with the simulation results. Moreover, we provided an in-depth analysis of the challenges arising when adapting VO algorithms for large FoV cameras, and adapted the state-of-the-art algorithm SVO to work with omnidirectional cameras.

Based on the simulations and experiments, it can be concluded that for small, confined environments, large FoV cameras should be used and for larger scale scenarios, small FoV cameras should be preferred.

References

[1] S. Ullman, *The Interpretation of Visual Motion*. MIT Press: Cambridge, MA, 1979.

[2] C. Tomasi and T. Kanade, “Shape and motion from image streams: a factorization method,” *Int. J. Comput. Vis.*, no. 7597, pp. 137–154, 1992.

[3] A. Chiuso, P. Favaro, H. Jin, and S. Soatto, “Structure from motion causally integrated over time,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 4, pp. 523–535, 2002.

[4] C. Forster, M. Pizzoli, and D. Scaramuzza, “Svo: Fast semi-direct monocular visual odometry,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 15–22.

[5] A. Rituerto, L. Puig, and J. Guerrero, “Comparison of omnidirectional and conventional monocular systems for visual SLAM,” *10th OMNIVIS with RSS*, 2010.

[6] B. Streckel and R. Koch, “Lens model selection for visual tracking,” in *Pattern Recognition*. Springer, 2005, pp. 41–48.

[7] A. Rituerto, L. Puig, and J. J. Guerrero, “Visual SLAM with an omnidirectional camera,” in *Int. Conf. on Pattern Recognition (ICPR)*, 2010, pp. 348–351.

[8] D. Scaramuzza and R. Siegwart, “Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles,” *Robotics, IEEE Transactions on*, vol. 24, no. 5, pp. 1015–1026, 2008.

[9] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis, “Monocular visual odometry in urban environments using an omnidirectional camera,” in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, 2008, pp. 2531–2538.

[10] D. Scaramuzza and F. Fraundorfer, “Visual odometry. Part I: The first 30 years and fundamentals,” *IEEE Robotics Automation Magazine*, vol. 18, no. 4, pp. 80 –92, Dec. 2011.

[11] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[12] P. Hansen, P. Corke, and W. Boles, “Wide-angle visual feature matching for outdoor localization,” *Int. J. of Robotics Research*, 2009.

[13] Z. Arican and P. Frossard, “Omnisift: Scale invariant features in omnidirectional images,” in *IEEE Int. Conf. on Image Processing (ICIP)*. IEEE, 2010, pp. 3505–3508.

[14] L. Puig and J. Guerrero, “Scale space for central catadioptric systems: Towards a generic camera feature extractor,” in *Int. Conf. on Computer Vision (ICCV)*, Nov 2011, pp. 1599–1606.

[15] M. Lourenço, J. P. Barreto, and F. Vasconcelos, “SRD-SIFT: Keypoint detection and matching in images with radial distortion,” *IEEE Trans. Robotics*, vol. 28, no. 3, pp. 752–760, 2012.

[16] P. Corke, D. Strelow, and S. Singh, “Omnidirectional visual odometry for a planetary rover,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, vol. 4. IEEE, 2004, pp. 4007–4012.

[17] D. Scaramuzza, “1-point-RANSAC structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints,” *Int. J. Comput. Vis.*, vol. 95, no. 1, pp. 74–85, 2011.

[18] S. Baker and I. Matthews, “Lucas-Kanade 20 years on: A unifying framework,” *Int. J. Comput. Vis.*, vol. 56, no. 3, pp. 221–255, 2004.

[19] J. Engel, J. Sturm, and D. Cremers, “Semi-dense visual odometry for a monocular camera,” in *Int. Conf. on Computer Vision (ICCV)*, 2013.

[20] R. Newcombe, S. Lovegrove, and A. Davison, “DTAM: Dense tracking and mapping in real-time,” in *Int. Conf. on Computer Vision (ICCV)*, Nov. 2011, pp. 2320–2327.

[21] J. Kannala and S. S. Brandt, “A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 28, no. 8, pp. 1335–1340, 2006.

[22] D. Scaramuzza, A. Martinelli, and R. Siegwart, “A flexible technique for accurate omnidirectional camera calibration and structure from motion,” in *Int. Conf. on Computer Vision Systems (ICVS)*. IEEE, 2006, pp. 45–45.

[23] C. Geyer and K. Daniilidis, “A unifying theory for central panoramic systems and practical implications,” in *Eur. Conf. on Computer Vision (ECCV)*. Springer, 2000, pp. 445–461.

[24] M. Lhuillier, “Automatic structure and motion using a catadioptric camera,” in *Proceedings of the 6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, 2005.

[25] L. Kneip and P. Furgale, “OpenGV: A unified and generalized approach to real-time calibrated geometric vision,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1–8.

[26] A. Pagani and D. Stricker, “Structure from motion using full spherical panoramic cameras,” in *Int. Conf. on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011, pp. 375–382.

[27] D. Scaramuzza, “Performance evaluation of 1-point-RANSAC visual odometry,” *J. of Field Robotics*, vol. 28, no. 5, pp. 792–811, 2011.