Lens Model Selection for Visual Tracking

Birger Streckel and Reinhard Koch

Institute of Computer Science and Applied Mathematics
Christian-Albrechts-Universität zu Kiel, 24098 Kiel, Germany

Abstract. A standard approach to generate a camera pose from images of a single moving camera is Structure From Motion (SfM). When aiming on a practical implementation of SfM often a camera is needed that is lightweight and small. This work analyses which is the best camera and lens for SfM, that is small in size and available on the market. Therefore we compare cameras with fisheye and perspective lenses. It is shown that pose estimation is improved by a fisheye lens. Also some methods are discussed, how the large Field of View can be further exploited to improve the pose estimation.

1 Introduction

Structure from Motion (SfM) is a well known approach for generating a camera pose from a single moving camera without the help of any markers [6]. A common application for SfM is the tracking of a persons head, i.e. in Augmented Reality where additional information has to be superimposed into the users view. Therefore the camera has to be attached to the users head and is possibly carried for a long time, hence a small and lightweight camera is inevitable.

This paper gives a detailed analysis on which is the best small and lightweight camera and lens configuration for SfM. Considered are cameras with fisheye and perspective lenses, it is shown that a camera with a fisheye lens is superior to a standard perspective camera. The presented results also apply to catadioptric cameras (perspective cameras looking into a convex mirror to get a hemispherical view), if they are designed to produce an image with constant angular resolution. These cameras are not suited for augmented reality, because the image center, commonly the users viewing direction, is blocked by the camera itself.

2 Previous work

Over the last years there was already some research done on which camera is best suited for SfM [3][7] [2][5][1]. The most extensive theoretical approach is made by Neumann and Fermüller in [3] and [7]. They state, that from theoretical considerations a fisheye camera is more appropriate for SfM than a single perspective camera, with a drawback because of the low resolution. There is no analysis done on how much the resolution deficit impairs the overall SfM Quality, instead a multi-camera is proposed with many perspective cameras looking in different directions and thus combining a high resolution with a big FoV.

There are different approaches showing that a wide FoV stabilizes the pose estimation, i.e. Davison stated in [2] that for perspective cameras with small

FoV, the motion along the optical axis is always ill defined because the camera moves towards the focus of expansion (FOE) (or focus of contraction (FOC)). Only the motion perpendicular to the FOE/FOC can be estimated reliably. In a spherical image with a wide FoV, there will always be many features with vectors having a large angle to the cameras optical axis, hence the estimation of the camera motion is always reliable.

3 Robust tracking from camera images

The implementation of the SfM approach [6] is divided into 2D feature tracking including initialization and robust 3D pose estimation.

Initialization and 2D feature tracking. In an initial step, salient 2D intensity corners are detected in the first image of the sequence. These 2D features are tracked throughout the image sequence by local feature matching, i.e. with the KLT operator [8]. Having corner correspondences between the first two images allows the computation of an Essential matrix between the views. With the Essential Matrix the relative pose of the cameras can be estimated and a 3D point can be triangulated for each 2D-2D pair.

3D feature tracking and pose estimation. After initialization each triangulated 3D feature point is assigned to a 2D feature track. From the third image on, the known 2D-3D correspondences can be used to determine the camera pose. Over time, new feature tracks are established and the 3D point positions are refined.

4 Lens Model Selection

In this section we will develop a simulation environment for the essential parts of SfM. This allows us to evaluate the effects of different lens models on point triangulation and camera pose errors.

Simulating Structure from Motion. The presented model implements the SfM approach for three images. As 3D scene a random 3D point cloud of 2000 points is generated, that is static for all experiments throughout this paper. The cloud is equally distributed in all directions around the camera center in a distance of 8-30 times the camera displacement length. Each camera can see parts of this cloud, these points are projected into the camera plane following the cameras projection model.

The only error source in SfM is the error in the feature position measurement of the 2D point tracker. If this tracker would return points with infinite accuracy and no wrong matches, the pose and scene estimation would also be perfect. According to [8] a good tracker is able to generate feature points with a standard deviation of $\sigma = 0.25$ pixel. To model the tracking error, Gaussian noise with a standard deviation of $\sigma = 0.25$ pixel is added to the points projection into the camera. With noise proportional to the pixel size, the tracking accuracy depends mainly on the camera resolution, which was fixed to $N = 1024$ in x- and y-direction. It is assumed, that all corner correspondences can be tracked independently of resolution and lens geometry and that no mismatches occur.

With the knowledge about point correspondences for the first two cameras, the pose of the second camera relative to the first can be estimated from the Essential matrix and 3D points are triangulated up to scale. Knowing estimated 3D points and the corresponding 2D projection, the pose of the third camera relative to the first can also be evaluated.

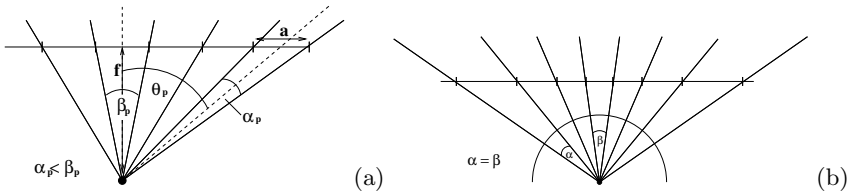

Fig. 1. Angular resolution of a perspective camera (a) and a fisheye camera (b).

Error models for perspective and equidistant projection. The difference between a perspective camera and a fisheye camera is the projection of a 3D point into the 2D image. The perspective camera performs a perspective projection, the ideal fisheye an equidistant projection [4]. These projections define the transformation of the modeled tracker noise into angular noise, which affects the triangulation quality.

An optimal fisheye lens gives a circular picture. The angle between the cameras optical axis and a ray through a 3D point (θ_p in fig. 1 (a)) is linear to the distance of the cameras focal point to the projection of the 3D point in the camera image. So for a fisheye the angular resolution is constant (see fig. 1(b)). In a perspective camera the angular resolution is higher for a greater angle θ_p, for points closer to the image border α_p is smaller than for points near the center (see fig. 1(a)).

To derive an error model for the perspective camera, it is necessary to calculate its angular resolution. From figure 1 (a) one can derive $\frac{f}{a}\sin(\alpha_p)+\sin(\frac{\alpha_p}{2}) = \cos^2\theta_p$. With $f >> a$ the term $\sin\frac{\alpha_p}{2}$ is negligible. It follows

$$\alpha_p \approx \text{asin}(\frac{a}{f}\cos^2\theta_p) = \text{asin}(\frac{2\tan(\theta_{max})}{N}\cos^2\theta_p), \tag{1}$$

where α_p is the angular resolution of the perspective camera, θ_p is the angle between the 3D point ray and cameras optical axis, f is the focal length, a is the size of a single CCD-Pixel, N is the full CCD-resolution and θ_{max} is the half FoV. With (1) we can compute the angular resolution of each pixel from its position on the chip.

The angular resolution of a fisheye camera is constant

$$\alpha_f = \frac{2\theta_{max}}{N}. \tag{2}$$

From (1) and (2) it is easy to see that the angular resolution of the fisheye camera is constant, while the angular resolution of the perspective camera is higher near the image borders. The functions for α_p and α_f are plotted in figure 2 for fisheye

and perspective cameras with different FoVs. Figure 2(a) shows α_p and α_f at angular image positions θ_p and θ_f, in figure 2(b) the angular resolution is plotted against the pixel positions on the CCD. The angular resolution for the center pixels is very bad for a perspective camera with a wide FoV, while for small FoVs fisheye and perspective cameras have a similar characteristic.

This leads to the result, that from the same Gaussian error on the pixel position follow two different models for the angular errors, which directly affect the quality of the scene reconstruction.

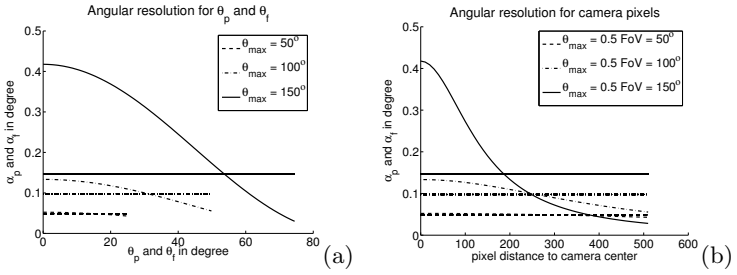

Fig. 2. Angular resolution for different FoVs.

Analysis of reconstruction and pose errors. The camera FoV has the greatest effect on the reconstruction error. For a camera with a constant number of CCD-pixels the cameras angular resolution declines with an increasing FoV. On the other hand the wide FoV increases the number of trackable features and also the features in view have a better spatial distribution which improves the pose estimation.

As stated above, points perpendicular to the FOE or FOC can be estimated most reliably. To simulate the effects of a changing FoV a critical camera movement was chosen. The first camera displacement is directly in z-direction, the FOE is in the image center. The second camera movement is in y-direction, perpendicular to the first movement.

From the first two images the camera displacement and rotation is estimated up to scale and 3D points are triangulated. The errors of the triangulated points are shown in figure 3(a) and (b). The curves give the mean of the relative triangulation error of 100 test runs on the same 3D point cloud with Gaussian noise applied for each projection. The triangulation results for x and y very similar and therefore represented by the same graph. They are better for a small FoV, because of the higher angular resolution of the camera. The z-coordinate estimation is always critical. The perspective camera z-error ascends fast with greater FoVs, because for a wide FoV perspective camera the angular resolution for center pixels gets very bad (see fig. 2(b)). The fisheye error characteristic is much better suited for triangulation at a wide FoV. The resolution degradation is compensated by the good estimation of points lying at the image borders, perpendicular to the FOE for the chosen movement. The z-error is therefore decreasing with increasing FoV.

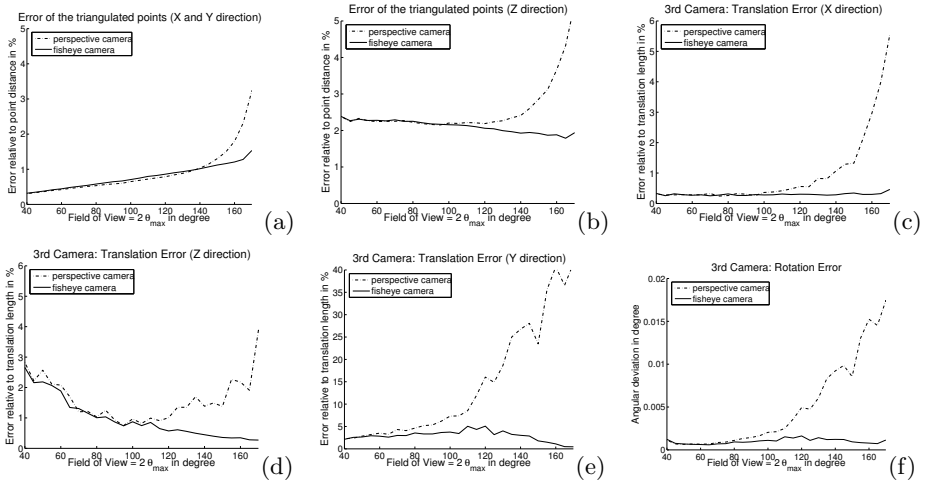

Fig. 3. (a), (b) Error of triangulated points from cameras 1 and 2 (z-displacement); (c)-(f) Error of predicted pose from camera 3 (y-displacement).

The third cameras pose is reconstructed from the estimated points and their known projection. The pose error is given in figure 3 (c) to (f). The third cameras pose prediction quality depends very much on the triangulated points quality (figure 3 (a) and (b)). Also number and spatial distribution of the points used for the estimation have an impact on the quality. It is visible that the y coordinate error is much higher, this is due to the chosen movement. The first camera move in z-direction produces high covariances in the 3D points z-coordinate. By the cameras second move in y-direction these are projected onto the camera images y-axis and cause a higher error in this direction.

The pose estimation quality for the third fisheye camera rises with the FoV, the additional points perpendicular to the FOE are more than compensating the lower resolution. The pose estimation for the perspective camera degrades fast, analog to the triangulated points quality.

We have simulated different motion directions for the first motion as well. All simulations showed a similar trend, that in all cases the fisheye lens performs equal or better than the perspective lens. See section 5 for further analysis. For a small FoV both models are similar, for a FoV $> 100°$ the fisheye lens is superior to the perspective lens.

Lens selection. From the preceding analysis follows that the precision of SfM is increased when using a fisheye lens with a large FoV. There are some further advantages of this lens that can hardly be modeled in this simulation but play an important part in a real system:

– The wide FoV covers a very large scene area and moving objects tend to be in a small part of the scene only. The hemispherical view will always see lots of static visual structures, even if the scene in front of the user may change dramatically.

– If fast rotations occur, a perspective camera will lose all tracked 2D features. This will not happen easily with a hemispherical view fisheye camera.

The drawback is, that the system is mainly designed for indoor use. In outdoor scenes the sun light falling directly onto the CCD sensor will cause problems. Also a cloudy sky would cause problems because a huge part of the scene is moving slowly enough to be still trackable. These problems can be facilitated by using CMOS sensors with logarithmic response and high dynamic range and a configuration where the fisheye camera is mostly looking to the ground.

5 Optimizing SfM for Spherical images

For a perspective camera there are certain critical moves for pose estimation. These are moves along the optical axis where the FOE lies close to the focal point. Points perpendicular to the camera movement are well suited for pose estimation. For critical moves with a small FoV camera most point vectors have small angles to the camera movement vector and thus the points have large triangulation errors. A full spherical camera that is able to see in all directions has no critical moves, since always all feature points are in sight. For a fair comparison between perspective and fisheye lens in section 4 we defined the pole of the fisheye hemisphere to coincide with the perspective optical axis. Standard SfM was than computed on a tangent plane with the optical axis as normal.

For a true spherical camera, we can select a tangent plane with a normal perpendicular to the FOE to avoid uncertainties. Since we do not know the camera movement and have to estimate it, we need a two step process. For FOE estimation we use the approach from section 4. In a refinement step we define an undistortion plane with its normal perpendicular to the FOE and use this for SfM. This is equivalent to a rotation of the fisheye pole to the plane perpendicular to the FOE. The result of this modification is shown in figure 4 for the triangulated points from the first two cameras (a),(b) as well as for the error of the third camera pose estimation (c)-(f). The camera movement is equal to section 4, first z-displacement then y-displacement.

The “virtually rotated fisheye” performs much better for a camera FoV of less than 130 degree but then rapidly degrades. This is due to the fact that the rotated small FoV camera mostly sees points perpendicular to the FOE with a very good covariance. But as the FoV angle grows, more and more points lying near the FOE or FOC are used for pose estimation. Since those points have a very bad covariance they spoil the whole pose estimation. In section 4 this effect is compensated by the good angular resolution of a small FoV camera, there the points close to the FOC are predominantly in view when the FoV is still small. Here more and more uncertain points are added while the angular resolution is decreasing. From figure 4 one can see the great influence of using the correct point distribution for pose estimation. It would also be useful to weight the estimates depending on their uncertainty w.r.t. the FOE.

From the results of this section follows that a virtually rotated fisheye with a FoV of approximately 100° outperforms a full fisheye as presented in section 4.

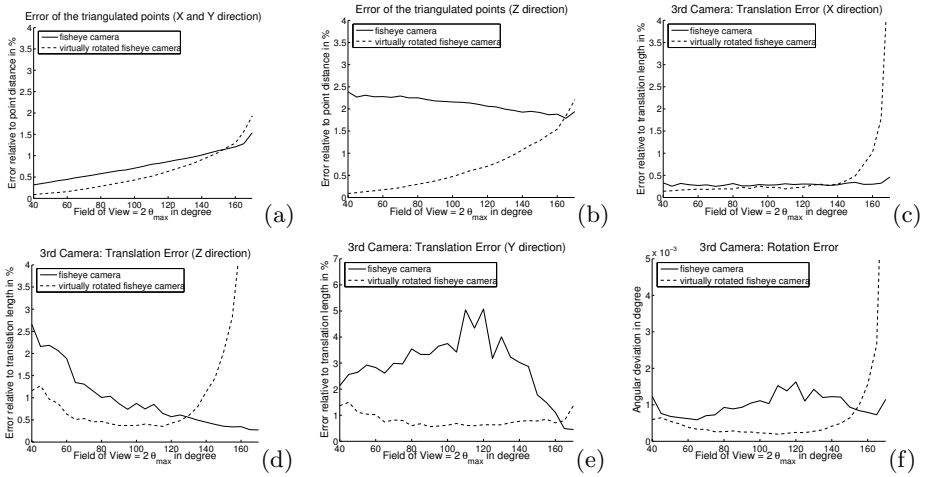

Fig. 4. (a), (b) Error of triangulated points from cameras 1 and 2 (z-displacement); (c)-(f) Error of predicted pose from camera 3 (y-displacement).

Knowing the FOE we could also perform a cylindrical rectification of the sphere aligned with the FOE direction. The cylinder would contain all tangent planes perpendicular to the FOE. This would further improve estimation quality, but was not done for this work.

6 Experiments

We have used perspective and fisheye sequences in an existing SfM system and evaluated the tracking results. To generate test sequences a high resolution (1600×1200 Pixel) fisheye camera with a FoV of 180° was used. Very accurate tracking on the full resolution images was performed, the results were taken as "ground truth" data. To compare the tracking performance on fisheye and perspective sequences, the centers of the high resolution images were mapped artifact free to perspective images of 400×400 Pixel with a FoV of 60° and an angular resolution of $6 - 8\frac{\text{pix}}{\circ}$. To simulate a fisheye camera with identical resolution a FoV of 160° was subsampled to 400×400 Pixel, the angular resolution was $2.5\frac{\text{pix}}{\circ}$. The results of the experiments are shown in table 1. These results are only partially comparable to the presented analysis, because in the used SfM implementation there are many additional stabilization methods and tracking was performed over several hundred images.

We have also verified the fisheye tracking over very long image sequences and have obtained very stable results. Figure 5 shows a video augmentation of virtual objects superimposed into the video stream in a complex environment.

	Translation Error	Rotation Error
Fisheye lens	3.3%	1.1%
Perspective lens	5%	1.4%

Table 1: Relative average translation and rotation error over 300 frames.

Fig. 5. (a) Original fisheye image, (b) and (c) video augmentation of a perspective cutout of the fisheye sequence from different view points.

7 Conclusions

The work shows that for SfM applications a fisheye lens is superior to a perspective lens, despite its lower average angular resolution. The lack of angular resolution is more than compensated by its linear characteristic. Also tracking is more robust because a greater part of the scene is in the cameras FoV.

There are still open issues to improve SfM with a fisheye camera. At first weighting of the 3D points w.r.t. the FOE for the camera pose estimation should be investigated. Points can be weighted according to their vectors angle to the cameras movement vector. Results from section 5 strongly imply that this is a promising approach. Furthermore real SfM on fisheye images without any projection has to be developed. The first step for this is already done in [9] where Svoboda et. al. evolved the epipolar geometry for fisheye images.

Acknowledgement: This work was supported by the Federal Ministry of Education and Research project ARTESAS (www.artesas.de).

References

1. P. Chang and M. Hebert. Omni-directional structure from motion. In *IEEE Workshop on Omnidirectional Vision 2000*, pages 153–160, 2000.
2. A. J. Davison, Y. González Cid, and N. Kita. Real-time 3D SLAM with wide-angle vision. In *Proc. IFAC Symp. on Intelligent Autonomous Vehicles, Lisbon*, July 2004.
3. Cornelia Fermüller, Yiannis Aloimonos, and Tomás Brodsky. New eyes for building models from video. *Computational Geom.: Theory and Applications*, 15:3–23, 2000.
4. M. Fleck. Perspective projection: the wrong imaging model. *Technical Report 95-01, Comp. Sci., U. Iowa*, 1995.
5. C. Geyer and K. Daniilidis. Catadioptric projective geometry. *Journal of Computer Vision*, 43:223–243, 2001.
6. R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge university press, 2000.
7. J. Neumann, C. Fermüller, and Y. Aloimonos. Eyes from eyes: New cameras for structure from motion. In *IEEE Workshop on Omnidir. Vision*, pages 19–26, 2002.
8. J. Shi and C. Tomasi. Good features to track. In *Conference on Computer Vision and Pattern Recognition*, pages 593–600, Seattle, June 1994. IEEE.
9. Tomás Svoboda, Tomás Pajdla, and Václav Hlavác. Epipolar geometry for panoramic cameras. *Lecture Notes in Computer Science*, 1406:218–??, 1998.