

Large-Field-of-View Stereo for Automotive Applications

Stefan K. Gehrig

Abstract—

For automotive applications, a 3D perception of the car's surroundings is crucial, both for driver assistance and for safety systems. In addition, a large field of view for applications such as intersection assistance.

A popular option to obtain 3D measurements of the surroundings is to use two passive cameras (stereo vision). Object detection is possible using this information. Automotive Applications based on object detection range from Adaptive Cruise Control, collision mitigation systems to intersection assistance systems.

Stereo vision with conventional cameras only delivers a limited field of view. We extend the field of view investigating active cameras (camera on a pan-tilt unit), panoramic cameras (camera and mirror), and fisheye cameras (camera with a fisheye lens).

The goal of this work is to find a setup that covers a large field of view and yields a good performance for object detection. The requirements for object detection are described and several options to meet these requirements are presented. We discuss their advantages and drawbacks and present results for all options.

Keywords— Stereo Vision, Active Vision, Omnidirectional Vision, Fisheye Lens, Intelligent Vehicle

I. INTRODUCTION

A. Perception Tasks for Automotive Applications

Modern vehicles need to be aware of their environment. This work wants to lay the basis for 3D measurements around the vehicle, especially keeping in mind the complex traffic scenario at intersections. A look at the accident statistics in most countries shows, that more than one quarter of all accidents occurs at intersections (see e.g. [1]).

Perception tasks for vehicles range from traffic light recognition, traffic sign recognition, lane recognition to traffic participant detection and recognition. For determining the desired properties of our stereo system, we focus on the object detection task.

Solving above tasks, many automotive applications can be implemented such as Adaptive Cruise Control, Stop and Go Driving, Speed Limit Assistance, Collision Mitigation and Intersection Assistance.

3D perception for vehicles is often performed by active sensors such as RADAR and LIDAR who obtain 3D measurements by time-of-flight measurements. Both options can obtain a large field of view but suffer from problems such as:

- RADAR sensors have difficulties in discriminating between objects on the ground and above.

- The measurement sensitivity depends heavily on the properties of the perceived material, e.g. a car yields far more echo than a human.
- Real-time LIDAR scanners use only one scan plane which can cause problems on hilly roads.

B. Stereo Systems

In this paper, we want to obtain 3D measurements of the surroundings with the use of two cameras (stereo vision). In that case, depth is extracted via triangulation, similar to human vision. Compared to monocular vision systems, this yields better performance due to the fact, that a model-free depth estimation for both static and moving objects is possible.

For an appropriate position of cameras in a car, very few places are deemed suitable. All forward-facing camera-based systems currently on the market are mounted in the wipeable area of the windscreen, often around the rearview mirror.

In this paper we limit ourselves to camera setups that are placed to the left and right of the rearview mirror. This way, vertical structures are better detected (favorable for traffic participants such as cyclists and pedestrians) and the cameras are placed in an area where the windshield wiper guarantees a good view. Cameras placed outside the wipeable area and facing forward exhibit problems with pollution. Cameras facing sideways are also an interesting packaging option for intersection assistance (in addition to cameras facing forward), but this setup is not investigated further here due to space limitations.

In this paper, we do not consider stereo systems that require very high mounting precision. Instead, we will calibrate the exact relative positions of the cameras w.r.t. each other offline with a software calibration step. Computation time for the calibration step is not relevant, however, the stereo computation of the stereo image pairs must be performed in real-time. Dedicated hardware exists for stereo computation but we focus on solutions that run on off-the-shelf PCs in real-time. Dedicated hardware is much less flexible than general purpose computers. A final consideration is cost, therefore we consciously use low-cost lenses and mirrors for these investigations.

Object detection at unsignalized crossings requires a large field of view in order to detect crossing objects early on (see e.g. [2] for initial approaches). We extend the limited field of view of conventional cameras in three ways:

- Active Cameras: Similar to the human driver using his eyes with a limited field of view, we use a conventional camera and mount it on a pan-tilt unit that allows us to inspect areas of our surroundings in (almost) any direction.

When we have to yield to vehicles coming from the right, we would turn our head in that direction. The same can be done with an active camera.

- **Panoramic Cameras:** Already popular in the field of robotics, we use a camera and a hyperbolic mirror to obtain a 360°-view of the scene. Since the projection properties are known, a rectified image can be obtained in any direction. However, the resolution of the scene is lower than with conventional cameras.
- **Fisheye Cameras:** A recent trend to low-cost fisheye sensors make them interesting for automotive applications. Already available as backing up aid, we try to use these lenses to perform metric measurements.

C. Measurement Requirements

For object detection in urban environments, a small child (size 30cm by 1m) is the most challenging object to detect. Keeping in mind the lower speeds in such environments, the object has to be detected at 30m distance, which corresponds to the braking distance from 50 km/h to 0 with 0.4g deceleration including some reaction time. Counter traffic has to be detected at larger distances but also has significantly larger dimensions, e.g. an approaching cyclist (size 1 by 2m) has to be detected at 40m distance. The frame rate of such systems has to be beyond 10 Hz for sufficiently fast responses.

D. Measurement Volume of Interest

For general considerations of accuracies the measurement volume of interest has to be defined. While forward moving, we want to monitor the area in front of our vehicle, preferably 180° horizontally. With the assumption, that we only need to respond to threats at speeds beyond 15 km/h, and that laterally intruding vehicles have a speed of 50 km/h maximum, we can set for a horizontal field of view of $2 \cdot \tan(50/15) \approx 150^\circ$. Vertically, we need a field of view of only 25°, which is roughly determined by the slope variations of the road and the geometry of the camera setting w.r.t. the car infrastructure (e.g. hood). So we need to monitor 25° by 150° in front of the vehicle.

How is this paper organized? In Section II the active cameras are described. Section III explains our omnidirectional approach with mirrors. The fisheye lens approach is detailed in Section IV. Results of the various hardware setups can be found in the respective sections. A final comparison and conclusions comprise the final section.

II. ACTIVE STEREO

A. Related Work

Active stereo setups have been described before in the literature for mobile robots (e.g. [3]). [4] presents an obstacle detection scheme for active stereo cameras. Some calibration procedures are mentioned, but no software calibration is necessary due to precise hardware alignment. [5] investigates the effect of Pan-Tilt-Zoom cameras being not centered in the optical center. The effect is small, but still not negligible for stereo computations.

B. Sensor Concept

Our demonstrator for an intersection assistant system is equipped with a digital stereo camera system, that is mounted on pan-tilt units (PTU)¹. The pan-tilt units are attached to the dashboard (see Figure 1). The camera system and the pan-tilt units are connected to the image processing computer, which is an Intel-based system. The pan-tilt units are basically servo/stepping motors that can turn the mounted cameras to (almost) any direction.



Fig. 1. Our PTUs in the demonstrator.

An alternative setup would be one pan-tilt unit with two cameras mounted on top. However, this setup is not acceptable when it comes to packaging issues. Single PTU-cameras can be packaged in small units whereas a Stereo-PTU automatically takes up at least the size of the stereo baseline.

C. Software Setup

Our approach to active stereo for automotive applications is as follows: We want to obtain 3D measurements from two cameras mounted on 2 PTUs. This requires a very precise determination of the pan-tilt axes. After a precise offline calibration of intrinsic and extrinsic parameters (see next subsection), GPS data with an attributed digital map delivers position data of an upcoming crossing and the PTU control steers the cameras to inspect the relevant area. 3D measurements are obtained for structured points with a standard stereo algorithm that needs standard stereo geometry (parallel pin-hole cameras with identical focal length). This geometry is obtained via image rectification, which is computed every frame using PTU position data and camera calibration data. From this 3D point cloud, a depth map (3D points viewed from a bird's eye view) is generated.

D. Calibrating an Active Stereo Camera System

For 3D measurements with a stereo-camera system the intrinsic and extrinsic parameters of the system must be known. These parameters are estimated by calibration. We use the algorithm of Jean-Yves Bouguet [6]. If the cameras are panned or tilted out of their initial position the extrinsic parameters (orientation and translation of the cameras

¹We use two Directed Perception PTU-D46-17 with 0.05° angular resolution.

to each other) will change unless a special arrangement is used, see for instance in [7]. If this isn't the case the pan- and tilt-axes must be calibrated in addition to the normal, static stereo-camera system.

For looking into an intersection it is not required to tilt the cameras, only a pan movement is needed. Therefore we now consider only the pan-axes. Both cameras are always panned with an identical angle to keep a maximum stereo field of view. A PTU-model is introduced characterizing the pan-axes and calculating new extrinsic parameters after a rotation of the cameras took place (see [8] for more details).

D.1 The PTU-model

Let \mathbf{R}_o be the rotation matrix denoting the orientation of the second camera w.r.t. the first one. The vector \mathbf{t}_o describes the translation of the second camera w.r.t. to the first one. \mathbf{R}_o and \mathbf{t}_o include the six extrinsic parameters obtained by calibration of the camera system in straight position (pan-angle = 0.0rad). The transformation of a point \mathbf{x}_{c2o} in the coordinate frame of the second camera into the first camera is given by:

$$\mathbf{x}_{c1o} = \mathbf{R}_o \cdot \mathbf{x}_{c2o} + \mathbf{t}_o \quad (1)$$

If the pan-axes are identical with the y_c -axes of the cameras, which would be ideal, only the position must be recalculated after a PTU rotation around the pan-angle α_{pan} . \mathbf{R}_o remains unchanged. The new translation \mathbf{t}_n is:

$$\mathbf{t}_n = \mathbf{R}(0, \alpha_{pan}, \mathbf{0}) \cdot \mathbf{t}_o \quad (2)$$

where $\mathbf{R}(0, \alpha_{pan}, \mathbf{0})$ is a rotation matrix representing a rotation around y -axis with the angle α_{pan} .

In general the pan-axes have arbitrary locations. Each is characterized by four parameters. These are two translation parameters t_{xa} and t_{za} (index "a" means axis) describing the displacement from the projection center of the camera. In fact it is the intercept point through the $x_c z_c$ -plane. The other two parameters are rotation parameters ω_a and κ_a describing the orientation (see Figure 2). All parameters are with respect to each camera coordinate frame.

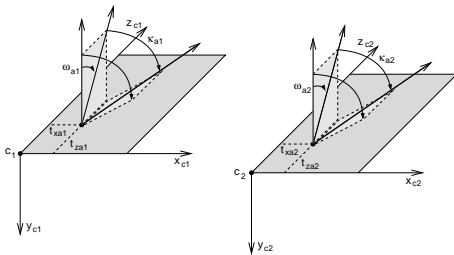


Fig. 2. PTU-Model. The pan-axes (red) are characterized by four parameters each. c_1 and c_2 represent the projection centers of the cameras. z_{c1} and z_{c2} are the optical axes.

If all pan-axes parameters are known the new extrinsic parameters (orientation \mathbf{R}_n and translation \mathbf{t}_n) caused by a rotation of the PTUs can be calculated. Two rotation matrices are introduced rotating around the pan-axes:

$$\mathbf{R}_{rot1} = \mathbf{R}_{a1}^T(\omega_{a1}, \mathbf{0}, \kappa_{a1}) \cdot \mathbf{R}_{pan}(\mathbf{0}, \alpha_{pan}, \mathbf{0}) \cdot \mathbf{R}_{a1}(\omega_{a1}, \mathbf{0}, \kappa_{a1}) \quad (3)$$

$$\mathbf{R}_{rot2} = \mathbf{R}_{a2}^T(\omega_{a2}, \mathbf{0}, \kappa_{a2}) \cdot \mathbf{R}_{pan}(\mathbf{0}, \alpha_{pan}, \mathbf{0}) \cdot \mathbf{R}_{a2}(\omega_{a2}, \mathbf{0}, \kappa_{a2}) \quad (4)$$

Equation number 3 rotates the first camera, where $\mathbf{R}_{a1}(\omega_{a1}, \mathbf{0}, \kappa_{a1})$ rotates first camera's pan-axis into the y_{c1} -axis. $\mathbf{R}_{pan}(\mathbf{0}, \alpha_{pan}, \mathbf{0})$ rotates the camera around the pan angle and $\mathbf{R}_{a1}^T(\omega_{a1}, \mathbf{0}, \kappa_{a1})$ rotates the pan-axis back. Equation 4 does the same with camera 2.

The new coordinates of a 3D point in coordinate frame of the first camera (\mathbf{x}_{c1n}) and the second camera (\mathbf{x}_{c2n}) are calculated as follows:

$$\mathbf{x}_{c1n} = \mathbf{R}_{rot1}(\mathbf{x}_{c1o} - \mathbf{t}_{a1}) + \mathbf{t}_{a1} \quad (5)$$

$$\mathbf{x}_{c2n} = \mathbf{R}_{rot2}(\mathbf{x}_{c2o} - \mathbf{t}_{a2}) + \mathbf{t}_{a2} \quad (6)$$

with $\mathbf{t}_{a1} = [t_{xa1}, 0, t_{za1}]^T$ and $\mathbf{t}_{a2} = [t_{xa2}, 0, t_{za2}]^T$.

The equations 5 and 6 are changed to \mathbf{x}_{c1o} respectively \mathbf{x}_{c2o} and put into equation 1 and solved for \mathbf{x}_{c1n} :

$$\begin{aligned} \mathbf{R}_n &= \mathbf{R}_{rot1} \mathbf{R}_o \mathbf{R}_{rot2}^T \\ \mathbf{t}_n &= -\mathbf{R}_{rot1} \mathbf{R}_o \mathbf{R}_{rot2}^T \mathbf{t}_{a2} + \mathbf{R}_{rot1} \\ &\quad \cdot (\mathbf{R}_o \mathbf{t}_{a2} + \mathbf{t}_o - \mathbf{t}_{a1}) + \mathbf{t}_{a1} \\ \mathbf{x}_{c1n} &= \mathbf{R}_n \mathbf{x}_{c2n} + \mathbf{t}_n \end{aligned} \quad (7)$$

The above PTU-model allows 3D measurements with an *active* stereo-camera system. The next section describes how to determine the pan-axes parameters. An extension of the model to host the tilt axis is straight forward.

D.2 Determination of PTU parameters

To determine the PTU parameters several approaches are available. In [3] for example a merged model including camera- and PTU parameters is established. With a laser pointer a huge virtual calibration target is generated. Already calibrated cameras measure the 3D position of the laser point. To calibrate the active stereo system the cameras must record many laser points in different panned positions.

In this paper the active stereo system is panned in a few positions. In every position the system is calibrated with Bouguet's algorithm. With the knowledge of the extrinsic parameters for these positions the PTU parameters can be calculated (intrinsic parameters remain constant). The PTU-model is nonlinear. The solution can be found iteratively by a gradient descent algorithm. The algorithm minimizes the following error function:

$$\sum_{i=1}^N \left\| \mathbf{R}_n^{(i)} - \hat{\mathbf{R}}_n^{(i)}(\mathbf{R}_{a1}, \mathbf{R}_{a2}, \alpha_{pan}^{(i)}) \right\| \cdot C_{weight} + \sum_{i=1}^N \left\| \mathbf{t}_n^{(i)} - \hat{\mathbf{t}}_n^{(i)}(\mathbf{R}_{a1}, \mathbf{R}_{a2}, \mathbf{t}_{a1}, \mathbf{t}_{a2}, \alpha_{pan}^{(i)}) \right\|^2 \quad (8)$$

Here the absolute value of a matrix means the sum of the squared elements. $\mathbf{R}_n^{(i)}$ and $\mathbf{t}_n^{(i)}$ denote the measured location of the cameras to each other in the i th pan position, while $\hat{\mathbf{R}}_n^{(i)}$ and $\hat{\mathbf{t}}_n^{(i)}$ denote the calculated location. The factor c_{weight} ensures a similar size between matrix sum and vector sum. This is only a crude cost function but yields very good results.

\mathbf{R}_{a1} , \mathbf{R}_{a2} , \mathbf{t}_{a1} and \mathbf{t}_{a2} include the eight PTU parameters to optimize. \mathbf{R}_{a1} and \mathbf{R}_{a2} are the rotation matrices introduced in Equations 3 and 4.

E. Rectification

For performing 3D measurements with our active camera system, we align the epipolar lines horizontally using a planar rectification (see [9]). We developed a fast rectification algorithm using the Graphical Processing Unit based on the programming language OpenGL. This algorithm puts hardly any computational burden on the CPU. With an NVidia GeForce 4 Ti 4600 the rectification of two 8bit gray value VGA images takes only 11ms. The algorithm takes the original image as a texture and binds it to a rectangle. Then the rectangle is placed within the OpenGL coordinate frame such that the image created by the OpenGL camera is the rectified image. Another advantage besides speed is the flexibility in changing the external orientation of the camera setup on the fly. Lookup-table-based approaches in software cannot cope with arbitrary PTU positions.

Alternatively, one could also use polar rectification, when the epipoles are not at infinity, which is the case in our application (see [10]). A combination of both methods is described in [11]. However, the rectification there is a two-stage process that still exhibits a problem, when both epipoles are at infinity. Since we do not want to perform stereo when one camera sees the other, we are not concerned with the situation of the epipole being in the image.

F. Stereo Computation Accuracy

Once standard stereo geometry is obtained, any stereo algorithm can be used to establish correspondences. We use a fast pyramidal approach [12] with the sum of squared differences (SSD) as correlation measure and subpixel disparity interpolation.

For measuring the accuracy a planar structured object, a traffic sign, was positioned parallel to the image plane. The measured depth (z_w) of the traffic sign was compared with the actual depth. The latter was provided by a laser pointer, whose error is assumed to be negligible. The depth ranges from 5 to 35m and the pan ranges from 0,0 (straight) to 50°(to the left). Figure 3 shows the absolute errors ($z_{w,camera} - z_{w,laser}$) of all single measurements.

In Figure 3(b) the error is the difference between the measured disparity of an object and its expected disparity. In the whole measurement range the error is smaller than one pixel. This accuracy is sufficient to detect objects in intersections and meets the requirements stated in Section I-C.

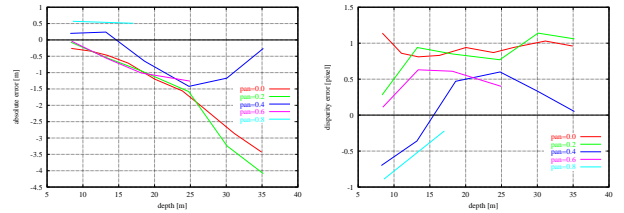


Fig. 3. (a) Absolute error of the active stereo rig versus object depth for several pan positions. The measurements were made intentionally at dusk (not ideal conditions for recording) to test robustness. (b) The graph shows the disparity error corresponding to the measurements.

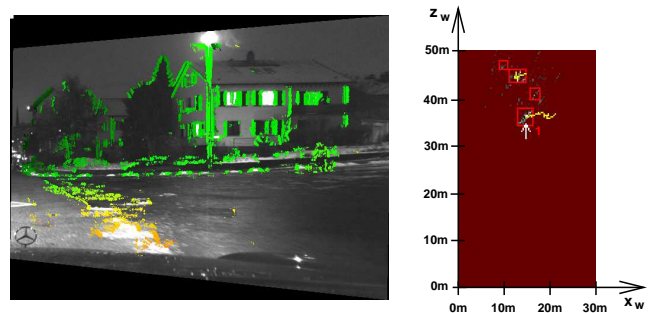


Fig. 4. Scenario with a bicycle at night. (a) A rectified image of the sequence with color-coded disparities. Red is near and green is far away. (b) The clustered depth map. The white arrow denotes a moving object. The trajectory of the object is marked with the yellow line.

We are mainly interested in horizontal accuracy for our application. The small field of view vertically has only little variation in resolution, independent of the camera type. For pinhole cameras the resolution depends on the view angle, the lowest resolution being in the center:

$$res_{pinhole}(\alpha)[pixel/^\circ] = res_0 \cdot \frac{1}{\cos^2(\alpha)}, \quad (9)$$

with $res_0 = f[pixel] \cdot \pi/180$ (f:focal length). α denotes the horizontal field of view. The effect of changing resolutions in a pinhole image is only apparent with fields of view beyond 60°, having a **higher** resolution at the edge of an image.

G. Cross Traffic Detection Results

The example (fig. 4) shows an intersection at night. Here a bicycle approaches the intersection. The cameras have a constant pan position of 0,4 rad to the right. The 3D position can be seen in the bird view perspective.

Other scenarios also with continuous panning motion show similar results (see [8] for more results).

H. Timings

The overall algorithm including object detection runs with approximately 11 Hz on 320*240 pixel images sufficiently fast (on a P4 3.2 GHz). The rectification of the VGA input images takes up 11ms and the stereo computation takes 60ms on rectified QVGA images.

I. Discussion

The active stereo approach delivers a resolution just like standard stereo configurations, only with a slight loss of accuracy towards the side due to the reduced net baseline. However, the field of view at one moment in time is still limited (40° in our example). The calibration stability (decalibration due to vibrations/motion over time) could be a problem due to the moving parts. With our setup we have a very high repeat accuracy of PTU motors and discovered little problems.

III. OMNIDIRECTIONAL STEREO

A. Sensor Concept

The use of an omnidirectional camera to obtain a large field of view is straightforward and has been used in many mobile robot applications. In an automotive setting suitable for serial production, we only consider the cameras to be mounted at the same position as conventional cameras, i.e. to the left and right of the rearview mirror.

B. Related Work

A plethora of work on omnidirectional cameras has been published in the last decade. [13] first published practical omnidirectional stereo results with a parabolic mirror. There, the hardware mounting was adjusted in such a way, that the epipolar lines turned out to be straight lines. In [14],[15], and [16] the epipolar geometry for catadioptric cameras is described in detail, including hyperbolic mirrors.

Omniscams have also been applied to intelligent vehicles. [17] shows a panoramic stereo setup for vehicles. Again, the cameras were precisely adjusted by hand. [18] does monocular omnivision image processing for 10m around the vehicle with inverse perspective mapping. In [12], initial steps of this work have been published.

C. Calibration

In this section we will mainly discuss the Calibration of the omnidirectional system. We start with a transformation of parts of the omnidirectional image into pinhole images. This can be done with a monocular image warping that needs to know the intrinsic parameters of the omniscam.

C.1 Organization of the Calibration of the System

The global calibration of the system can be decomposed into three different steps:

1. **Mono-Calibration:** During this step we consider the two cameras without connection between them: we just compute the parameters of the hyperbolic mirror (ϵ , curvature (= focal length), principal point).
2. **Pinhole Transformation:** With the parameters from the previous step we can generate a classic pinhole image, i.e we transform the omni image into an image without distortion due to the hyperbolic mirror.
3. **Stereo-Calibration:** Finally in this step we take as input the left and right pinhole images and we compute the

stereo parameters of the system, i.e the matrix (composed by a translation vector and a rotation matrix) which transforms the frame linked to camera 1 into the frame linked to camera 2. By knowing that matrix we can do stereo rectification and computation.

The three steps could be combined to one step. The stereo results for the relevant image parts are optimized for that specific part of the omni image, which is better than an optimization of the full omni image.

With direct stereo of the omniscam images, one has to take the warping of image patches into account before correspondence analysis can take place. For more details about the epipolar geometry of omniscams, consult e.g. [19].

C.2 Mono Calibration of a Hyperbolic Omnicamera

To calibrate the camera, we image a black and white chessboard target at various orientations and locations and extract the corners of the chessboard.

After this step we compute the following parameters:

- the camera position in world coordinates: $\alpha, \beta, \gamma, t_x, t_y$ and t_z . So in other words we estimate R and T , known as extrinsic parameters.
- the internal camera parameters: focal length (f), principal point C_u, C_v
- the eccentricity of the hyperboloid (ϵ)
- the chessboard center $grid.x_0$ and $grid.z_0$

The algorithm performs a bundle adjustment and we obtain a 12-dimensional vector. Even if we have an initial vector which is close to the solution we can have a false solution because of local minima. One solution is to take into consideration fewer parameters. We can set the other parameters with realistic values by measuring the position of the omniscam w.r.t. calibration grid.

In theory the stereo calibration step could be incorporated in this model but we did not for convenience reasons.

C.3 Omnicam Model

As explained in the previous section, we use the camera model of Bouguet, so we have to take into account the radial lens distortion contribution. The effects of the radial distortions are not negligible (we neglect effects of affinity and shearing and higher order lens distortions). Let ϵ_D be the vector of the lens distortion.

So in our model estimation, we have the following equality:

$$\epsilon \hat{=} \epsilon_{hyper} + \epsilon_D \quad (10)$$

As we optimize all omniscam parameters in the lens-mirror system we will not be able to obtain the value of the distortion coefficient. We can just obtain ϵ and not ϵ_{hyper} .

With this simple model, we assume that the mirror center and the center of the lens distortion are identical.

The quality of our reduced model can be estimated by the reprojection error. With an RMS of roughly 0.6pixel, our model is sufficiently accurate.

C.4 Pinhole Transformation

The process to create a pinhole image from a distorted image is known as rectification. This pinhole image repre-

sents the output image of a pinhole camera. It is referred to as a virtual pinhole camera. The idea behind the pinhole Transformation is to convert the original omniscam image into a virtual pinhole image.

The remaining problem is to compute the mapping $(x', y') \rightarrow (x; y)$ from 2D virtual pinhole coordinates to 2D Omnicam coordinates. This mapping is well defined, if the virtual pinhole camera and the Omnicam camera have the same center of projection, which we choose here.

One other important design choice is the configuration of the virtual pinhole images. By configuration, we mean the angles for the different viewing directions. We want to monitor the scene through three different directions: center, left and right direction. This reduces the effect of changing resolutions due to Equation 9 significantly. One image with a field of view of around 150° would yield similar results with less resolution in the center image part.

We chose a field of view of $60^\circ \times 30^\circ$ for one virtual pinhole image. We have a portion of the 3D space which is included in two different viewing directions. We choose the viewing directions to be $\pm 45^\circ$ apart, so we will have a 15° overlap. We need a little overlap for the Stereo Rectification to prevent blind spots.

C.5 Stereo Calibration of a Hyperbolic Omnicameras System

We use the algorithm of Zhang (see [20]) and Bouguet (see [6]) as described in the previous section. We want to use this method for calibrating the extrinsic parameters of our stereo system.

The dimensions of the image are set to 400×200 . This reflects the fact that we have an initial omni image of 1024×768^2 . So in order to minimize interpolation effects, the best choice is 400×200 . Some calibration parameters of the virtual pinhole image are known by construction:

- Due to the virtual pinhole model, we have, by definition, kept the center in the middle of the Image.
- perfect pinhole model means no radial distortions and no shearing ($K_1 - K_5 = 0$).
- Because of the virtual pinhole model, we know (FC_u, FC_v) . We have a FOV such as $60^\circ \times 30^\circ$, so this yields focal length FC:

$$(FC_u, FC_v) = \left(\frac{400}{2 * \tan 60^\circ / 2}, \frac{400}{2 * \tan 30^\circ / 2} \right) \quad (11)$$

We used the MATLAB Bouguet implementation to perform the Stereo calibration [6].

If we consider the system in the configuration for the left stereo calibration (i.e with $\beta = -45$), we just need to apply a transformation to the couple (R_{center}, T_{center}) in order to obtain the couple (R_{left}, T_{left}) and (R_{right}, T_{right}) .

D. Rectification

Having performed a stereo calibration with the virtual pinhole images, we can use this information to rectify the

²We use a MicroPix CCD M 1024 with a VStone hyperboloid mirror.

virtual pinhole images using planar rectification ([9]). The rectified views overlaid with stereo results are shown in Figure 6 bottom. Above one can see the virtual pinhole images.

E. Stereo Computation

With the preparations described above, for all 3 views stereo can be computed. See Figure 5 for an example. The distorted appearance of the left and right view are due to the planar stereo rectification, similar to our active stereo setup at 45° pan position. The views are computed for $(-45^\circ, 0^\circ$ and $45^\circ)$.



Fig. 5. Stereo computation result with 3 views of a traffic scene. Note the slight overlap of the 3 views.

The 3D position accuracy is significantly lower than with conventional stereo cameras. The main reason for that is the lower resolution. Also, the calibration procedure does not model the hyperbolic stereo setup precisely.

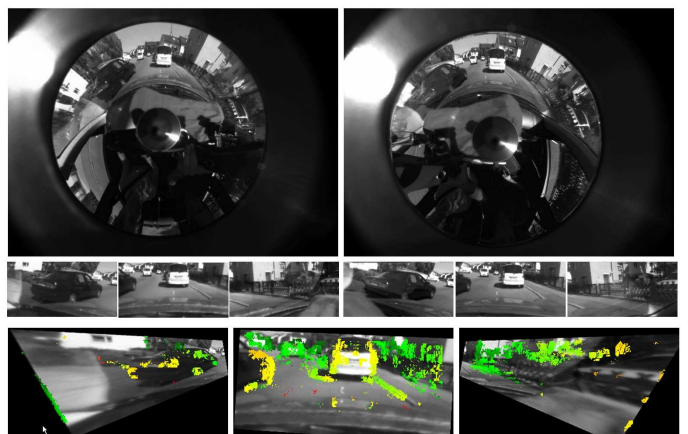


Fig. 6. Stereo computation result with 3 views of a traffic scene. The small views underneath the omni images are virtual pinhole images from above omniscam image. Disparities are color-coded like in Figure 4.

F. Accuracy

We used two 1024×768 camera systems for hyperboloid stereo. For omniscams the resolution is independent of the

viewing angle, so $res_{omni}(\alpha) = res_0$. With optimal imaging of the mirror, one can obtain a maximal horizontal resolution of $2 \cdot \pi \cdot 384pixel/360^\circ \approx 6pixel/^\circ$. On our setup, we could not cover the full image height with the mirror image and the outermost part of such omniscam images do not obey the mirror geometry perfectly, so we obtain a $4.5pixel/^\circ$ resolution. This yields roughly a factor of 3 lower accuracy than our active stereo setup and motivated the choice for our virtual pinhole image size of 400 by 200. Comparison with ground truth (a laser range finder) shows, that the accuracy is within 10cm for ranges up to 8m (see table below). A small child at 30m covers 3 by 10 pixels and yields a disparity of 3 pixels, which makes a robust detection very hard.

The vertical resolution of mirror systems varies significantly being the highest at the outermost part of the image, but since we only use a 30° field of view vertically, this is not a large problem.

Direction	x_{las} (m)	x_{res} (m)	z_{las} (m)	z_{res} (m)
center	0.62	0.70	1.75	1.69
center	-1.22	-1.18	2.17	2.38
center	1.07	0.95	8.57	8.12
center	0.37	0.44	2.13	2.20
center	σ_{err} (cm)	8.8	σ_{err} (cm)	16.8
left	1.76	1.81	2.27	2.32
left	6.98	6.61	3.25	3.10
left	2.68	2.77	1.37	1.52
left	0.82	0.71	1.59	1.43
left	7.32	7.14	4.45	4.27
left	σ_{err} (cm)	15.2	σ_{err} (cm)	13.8
right	-0.47	-0.59	1.58	1.66
right	-0.53	-0.54	1.62	1.55
right	-0.68	-0.77	2.13	2.20
right	-0.62	-0.73	1.49	1.37
right	σ_{err} (cm)	8.2	σ_{err} (cm)	9

TABLE I

COMPARISON BETWEEN STEREO RESULTS AND MEASURED LASER POINTS (σ_{err} = MEAN ERROR).

G. Timings

Mono-Rectification of the omniscam images takes 15ms (six 400x200 views), planar stereo rectification for these 3-2 views 20ms, and stereo computation for these 3 image pairs 45ms, resulting in a frame rate of roughly 10 Hz. These measurements were taken on a 1.7 GHz P4 Pentium machine.

H. Discussion

With omniscams, we obtain the full field of view at all times, however, at a significantly lower resolution than with conventional cameras. Good imaging quality with well manufactured mirrors can be obtained, the long-term calibration stability is more critical than with conventional cams, since the mirror-lens-mounting introduces more degrees of freedom. The image area is not efficiently used, since only 180 degrees can be used for environment perception, unless one wants to perform driver monitoring at

the same time. Due to the lower resolution, the object detection requirements are difficult to meet.

IV. FISHEYE STEREO

A. Sensor Concept

With fisheye lenses on conventional cameras, a large field of view is obtained. The resolution per view angle is constant for well-designed fisheye lenses, so $res_{fisheye}(\alpha) = res_0$.

B. Related Work

Fisheye lenses have become increasingly more popular in Computer Vision due to the reduced costs. [21] shows an approach of calibrating fisheye lenses using planes. Only radial distortions are considered which are by far the most dominant aberrations for fisheye lenses. Several distortion models are introduced. A more thorough overview of camera models for wide angle lenses can be found in [22].

C. Calibration

Calibrating fisheye lenses can be done similar to Bouguet's distortion estimation algorithm, only the camera model differs. We tried both the polynomial model with 4 distortion coefficients and the division model with 4 coefficients (see [21] for model details). The differences were very small. The distortion center was determined manually and with the measured corner positions of our calibration pattern the distortions coefficients were determined via least squares. These undistorted fisheye images are input to the Bouguet stereo calibration. With this setup, the distortions are set to zero and the principal point is known.

D. Rectification

The images in our example were taken with VGA firewire cameras and low-cost fisheye lenses³ with a field of view of roughly 150° . For our setup with 150° field of view, a resolution of $4.5pixel/^\circ$ is obtained, similar to the omniscam image but with a lower imager resolution. An example is shown below.

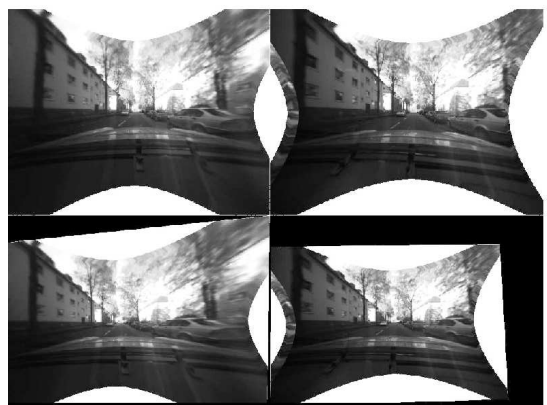


Fig. 7. Undistorted fisheye images (top), and the same images rectified (bottom). Note the smaller field of view in the right image due to the larger focal length of the right lens.

³Bowoon 216050CC fisheye lens

E. Stereo Computation

With above preparation steps, it is again straightforward to compute stereo correspondences. Below is a stereo result of the rectified image pair from Figure 7.

In order to speed up computation, one could perform a pinhole projection in y-direction and a cylindrical projection in x-direction. This way the image size in x-direction always stays bounded, even for fisheye lenses with a field of view greater than 180° . Also, the angular resolution remains constant and the simple epipolar geometry remains. Only the 3D computation becomes dependent on the image position which introduces little computational burden.

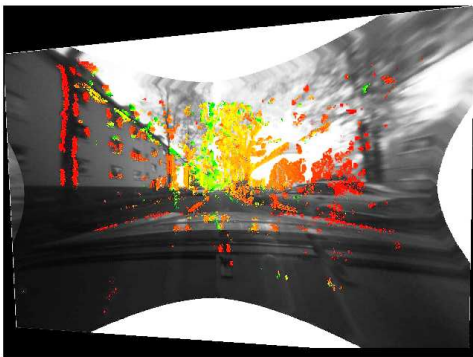


Fig. 8. Stereo result of above fisheye images. Due to different camera brightness control and low contrast in many regions, occasional erroneous correspondences occur (contrast stretching applied for display purposes). Disparities are color-coded like in Figure 4.

F. Timings

Detailed timings have not been conducted yet but since the fisheye camera model is applied via lookup just like the Bouguet model, no computational difference is expected.

G. Discussion

Fisheye lenses show a moderate image resolution with usable large portions of the image. In our setup, we would obtain a resolution of $7\text{pixel}/^\circ$ when using a 1024×768 imager.

Problems can be a noticeable loss of the non-single-viewpoint property, chromatic aberrations, and different focal lengths for different viewing angles. Especially low-cost fisheye lenses exhibit these problems. Calibration stability over time is not a problem, since these camera setups are comparable to conventional cameras.

V. COMPARISON AND CONCLUSIONS

A comparison of options to compute stereo for a large field of view for automotive applications has been presented. An active stereo camera system exhibits the highest accuracy for such a task at the cost of not having a large field of view at one instant in time. Stereo omniscam cameras and stereo cameras with fisheye lenses offer a complete field of view at any time but have a lower image

resolution and consequently a lower 3D measurement accuracy. With the same imager resolution, the fisheye stereo setup uses the imager area more thoroughly and yields a better resolution. The fisheye stereo setup also exhibits the best calibration stability.

REFERENCES

- [1] Bundesamt für Statistik, *Verkehr in Zahlen 2000 (Datengrundlage 1999)*, Deutscher Verkehrs-Verlag, 2000.
- [2] S. K. Gehrig, S. Wagner, and U. Franke, "System architecture for an intersection assistant fusing image, map, and gps information," in *Proceedings of the Intelligent Vehicles 2003 Symposium, Columbus, Ohio, USA*, June 2003.
- [3] J. D. Davis and X. Chen, "Calibrating pan-tilt cameras in wide-area surveillance networks," in *Proceedings of Int. Conference on Computer Vision 03*, 2003.
- [4] F. Li and M. Brady, "Modeling the ground plane for real-time obstacle detection," *Computer Vision and Image Understanding*, vol. 71, no. 1, pp. 137–152, 1998.
- [5] E. Hayman and D. W. Murray, "Effects of translational misalignment when self-calibrating rotating and zooming cameras," *IEEE Transact. Pattern Analysis & Machine Intelligence*, vol. 25, no. 8, pp. 1015–1020, 2003.
- [6] J.-Y. Bouguet, *Camera Calibration Toolbox for Matlab*, Caltech, 2000, Available at <http://www.vision.caltech.edu/bouguetj/calib4oc/index.html>.
- [7] Y. Yoshikawa, Y. Tsuji, M. Asada, and K. Hosada, "View-based imitation with rotation invariant pan-tilt stereo cameras," in *International Conference on Intelligent Robots and Systems*, 2002.
- [8] S. K. Gehrig, J. Klappstein, and U. Franke, "Active stereo intersection assistance," in *Vision Modeling and Visualization Conference, Stanford, USA*, November 2004.
- [9] A. Fusiello, E. Trucco, and A. Verri, "Rectification with unconstrained stereo geometry," in *British Machine Vision Conference*, 1997.
- [10] M. Pollefeys, R. Koch, and L. Van Gool, "A simple and efficient rectification method for general motion," in *Proceedings of Int. Conference on Computer Vision 99*, 1999, pp. 496–501.
- [11] D. Oram, "Rectification for any epipolar geometry," in *British Machine Vision Conference*, September 2001.
- [12] U. Franke, S. K. Gehrig, and F. Lindner, "Camera-based intersection assistance," in *Aachen Colloquium Automobile and Engine Technology*, October 2004, pp. 803–820.
- [13] J. Gluckman, S. K. Nayar, and K. J. Thoresz, "Real-time omnidirectional and panoramic stereo," in *Image Understanding Workshop DARPA*, 1998.
- [14] T. Svoboda and T. Pajdla, "Epipolar geometry for central catadioptric cameras," *IJCV*, vol. 49, pp. 23–37, 2002.
- [15] T. Svoboda, T. Pajdla, and V. Hlavac, "Epipolar geometry for panoramic cameras," in *European Conference on Computer Vision (ECCV)*, 1998, pp. 218–232.
- [16] S.-K. Wei, "Stereo matching of catadioptric panoramic images," Tech. Rep., Czech Technical University, Report No. CTU-CMP-2000-08, 2000.
- [17] L. Matuszyk, A. Zelinsky, L. Nilsson, and M. Rilbe, "Stereo panoramic vision for monitoring vehicle blind-spots," in *Proceedings of the Intelligent Vehicles 04 Symposium*, 2004, pp. 31–36.
- [18] T. Gandhi and M. Trivedi, "Motion based vehicle surround analysis using an omnidirectional camera," in *Proceedings of the Intelligent Vehicles 2004 Symposium, Padova, Italy*, June 2004.
- [19] C. Geyer and K. Daniilidis, "Structure and motion from uncalibrated catadioptric views," in *Proceedings of Int. Conference on Computer Vision and Pattern Recognition 01*, 2001.
- [20] Z. Zhang, "A flexible new technique for camera calibration," Tech. Rep., Technical Report MSRTR-98-71, Microsoft Research, 1998.
- [21] S. Thirtala and M. Pollefeys, "The radia trifocal tensor: A tool for calibrating the radial distortion of wide-angle cameras," in *Proceedings of Int. Conference on Computer Vision and Pattern Recognition 2005*, 2005.
- [22] Luhmann, *Nahbereichsphotogrammetrie*, Springer Verlag, 2000.