Topview Stereo: Combining Vehicle-Mounted Wide-Angle Cameras to a Distance Sensor Array

Sebastian Houben

Institute for Neural Computation, University of Bochum, Germany

ABSTRACT

The variety of vehicle-mounted sensors in order to fulfill a growing number of driver assistance tasks has become a substantial factor in automobile manufacturing cost. We present a stereo distance method exploiting the overlapping field of view of a multi-camera fisheye surround view system, as they are used for near-range vehicle surveillance tasks, *e.g.* in parking maneuvers. Hence, we aim at creating a new input signal from sensors that are already installed.

Particular properties of wide-angle cameras (*e.g.* changing resolution) demand an adaptation of the image processing pipeline to several problems that do not arise in *classical* stereo vision performed with cameras carefully designed for this purpose. We introduce the algorithms for rectification, correspondence analysis, and regularization of the disparity image, discuss reasons and avoidance of the shown caveats, and present first results on a prototype topview setup.

1. INTRODUCTION

Modern vehicles possess a continuously growing number of sensors to allow for assistance in and possibly avoidance of difficult and dangerous situations. In the long term, vehicles will also rely on their sensors to perform fully autonomous maneuvers and rides.

Among these sensors are ultrasound, video cameras, lasers, LIDAR and similar. They differ in resolution, range, detectable material and computational effort for postprocessing their outputs. This variety substantially drives the cost of modern vehicles.

We present a stereo distance sensor that is based on the combination of the camera array used for a vehicle surround view, aka topview. Up to date topview camera systems are used in difficult maneuvering situations like parking or near-range obstacle avoidance. This is usually achieved by displaying a surrounding view to the driver. Recent research also shows possibilities of automating the recognition of the environment, *e.g.* the search for free parking lots.[1]

In the following, we introduce a method to automatically compute a rectified stereo image pair for two overlapping cameras from the topview system (*cf.* Sec. 3). We present the arising difficulties for stereo correspondence matching (*cf.* Sec. 4) and ways to handle them, show results on a topview system (*cf.* Sec. 6) consisting of four fisheye cameras, and close the paper with a discussion of the advantages and caveats of this new stereo sensor and derive situations where they can be effectively deployed (*cf.* Sec. 7).

2. RELATED WORK

The literature on real-time stereo vision is vast and profound. We restrict ourselves to pointing out the subjective key papers in this field with regard to driver assistance systems. The general line of action consists of three steps:

- Rectification: Computing a rectified image pair, *i.e.* transformations for the input images of the used stereo cameras that maps corresponding points to the same image line and, thus, facilitates correspondence analysis

E-mail: sebastian.houben@ini.rub.de, Telephone: +49 (0)234 - 32 - 25566

- Matching cost: For every pair of pixels lying in the same row of the two different images below a given disparity, a dissimilarity measure is computed, yielding a three-dimensional cost array with the coordinates row, column and disparity
- Accumulation: The cost array is processed into a two-dimensional disparity image where each pixel is assigned the disparity to a corresponding pixel in the other rectified stereo image. Usually some smoothness assumptions are imposed during this process.

Image rectification for pinhole cameras (or cameras that can be sufficiently modeled by a central projection) is extensively attended to in the standard literature (*cf.*[2]). In the following, we will refer to this procedure as *classical* rectification. For the geometric intuition of the proposed fisheye rectification we want to refer to Pollyfeys et al.[3] who found a very comprehensible method to transform images from a camera setup with generic geometry to a rectified image pair. Likewise, the calibration of fisheye stereo camera pairs is covered in[4] from a more theoretical standpoint.

The correspondence matching, being the computationally most expensive part, is central in every real-time stereo implementation. We refer to Hirschmüller[5][6] for a survey of cost functions for different purposes and video sensor properties.

Uncounted methods have been proposed for computing a disparity map from the correspondence cost. With regard to real-time applicability we cite the papers by Forstmann et al.[7] engaging in Dynamic Programming and the widely-used Semi-Global Matching approach[8][9]

For a reader new to the field we would like to recommend the well-established KITTI Benchmark Dataset[10] to gain a quick overview over state-of-the-art methods.

3. RECTIFICATION

To facilitate the correspondence analysis it is imperative to compute a pair of rectified images, *i.e.* a transform of the input images from both stereo cameras which ensures that corresponding points lie in the same image line. This condition is known as the epilolar constraint. The width w and height h of the rectified images can be chosen arbitrarily.

The classical rectification consists of determining two homographic mappings estimated from a set of corresponding points in both input images. This procedure is described in detail in[2] and is the method of choice for cameras that can be modeled by a central-projection to an image plane. However, cameras with strong lens distortions like wide-angle and fisheye cameras that are usually deployed in topview systems are not feasible for this approach, since undistorting the input image often introduces strong numerical instabilities. Furthermore, the classical model does assume a near-parallel alignment of the cameras which is not given in a topview system.

We set up the epipolar constraint in three dimensions and derive the pixel-wise transform automatically from the relative alignment of the regarded camera pair with a single viewpoint that share an up to 180° overlapping field of view. The alignment is arbitrary except for the presumption that no camera center is located inside the field of view of any other (*cf.*[3] for this special case). With respect to the field of view we may then speak of a *left* and a *right* camera. For the later correspondence analysis it is also important to note that our approach will guarantee an order within the lines of the rectified images: a pixel in the right rectified image will always correspond to a pixel in the left rectified image that is located right of its original position. Due to its use of sphere coordinates to sample epipolar curves on the distorted images, we refer to this method as *spherical rectification*.

3.1 Spherical rectification

We assume that the position $c_{R/L}$ and orientation $R_{R/L}$ of both cameras respectively within the vehicle coordinate system is given. We denote the camera orientations by the orthogonal 3-by-3 matrices $R_{R/L}$ that rotate a point in world coordinates into the camera coordinate system. Since we regard this problem in the context of a vehicle-based camera system, we also align the stereo view parallel to the ground plane, thus, making use of its normal vector n_G. Furthermore, we presume knowledge of the lens distortion functions $d_{R/L} : \mathbb{R}^2 \to \mathbb{R}^3$ mapping an

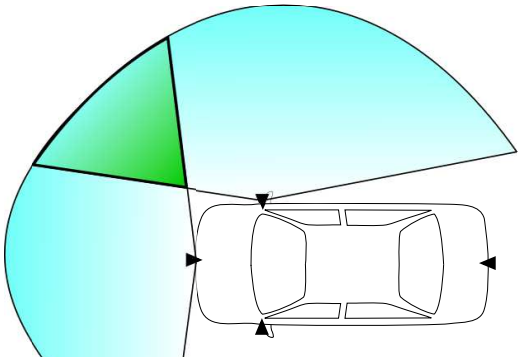


Figure 1. The problem at hand: A vehicle is equipped with four wide-angle cameras provides a surround view. The overlap between any two of the cameras is suitable for stereo vision.

image point in 2D to its corresponding view ray emanating from the camera center. Lastly, a binary image masking lens border regions within the image is needed.

We suggest to follow the explanation via Fig. 2. For determining a useful field of view we start with the view ray $e = (0, 0, 1)^T$ pointing directly forward in the camera coordinate system and project it to the ground plane yielding v_F.

$$v_F := R_L^T e - \langle R_L^T e; n_G \rangle n_G$$
$$v_F := sgn(\langle v_F; e \rangle) v_F$$

where the second assignment makes sure that v_F is still pointing in view direction. Let v_c be the vector between both camera centers

$$v_c = c_R - c_L$$

In order to find the boundaries of the field of view in the plane spanned by v_F and v_C we now sample $R_{sgn(\langle v_F, v_C \rangle)\alpha} v_F$ for angles α where R_ϕ denotes the rotation around $v_F \times v_C$ with the angle ϕ. The largest and smallest of these angles where

$$d_R^{-1}(R_R R_{sgn(\langle v_F, v_C \rangle)\alpha} v_F) \text{ and } d_L^{-1}(R_L R_{sgn(\langle v_F, v_C \rangle)\alpha} v_F)$$

still yield a valid backprojection (a coordinate within the image bounds and masked by the aforementioned image sensor mask) in both cameras donates the vectors v_R and v_L, the outer boundaries of the overlapping field of view.

In order to now map a pixel (r, c) of the wanted rectified image to its correspondent coordinates in both camera images, we divide the overlapping field of view into w angle sections (by choosing the width of w pixels of the rectified image we, thus, choose an angle resolution of $\theta°$/pixel as well). Likewise, by choosing h we obtain a vertical opening angle of $h\theta°$.

Next, we rotate v_L around v_C with angle $r\theta$ and the result v_B around $v_L \times v_C$ with angle $c\theta$ to obtain the view ray v_{RC}. Finally $d_L^{-1}(R_L v_{RC})$ and $d_R^{-1}(R_R v_{RC})$ yield the wanted input image coordinates.

Please note: v_{RC} emanating from c_L and v_{RC} emanating from c_R do not intersect. Additionally, v_{RC} splits the epipolar plane, *i.e.* the plane spanned by v_c and v_{RC}, into two half-spaces. Every scene point in front of the camera pair lying on $c_L + \lambda v_{RC}, \lambda > 0$ can be obtained by an intersection with $c_R + \mu v, \mu > 0$ where v lies in the left half-space.

Inversely, it is possible to compute the coordinates of a given view ray v_W in one of the rectified images. For this, we project v_L to the epipolar plane, spanned by v_W and v_c, to obtain the leftmost view ray on that plane v_{LL}. The vertical angle between v_L and v_{LL} and the horizontal angle between v_{LL} and v_W now yield the row and column coordinate of the rectified image respectively.

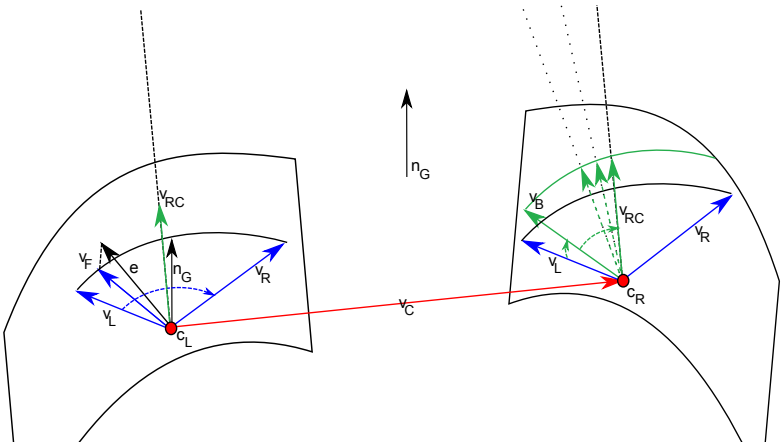


Figure 2. The performed geometric computations: The boundaries of the overlapping field of view are sampled (blue), for a given rectified pixel coordinate (r, c) the vectors v_B and v_{RC} are determined (green). All scene points located on v_{RC} emanating from c_L in front of the stereo camera pair intersect with view rays from c_R rotated left from v_{RC}

3.2 Properties

In contrast to classical projective cameras a pixel of the proposed rectified image describes a solid angle. This agrees to the properties of wide-angle cameras where, in particular in the image's border area, a distance in the image is nearly proportional to its view angle. As a consequence, there is no simple relation between a point's distance and its disparity in the stereo image pair.

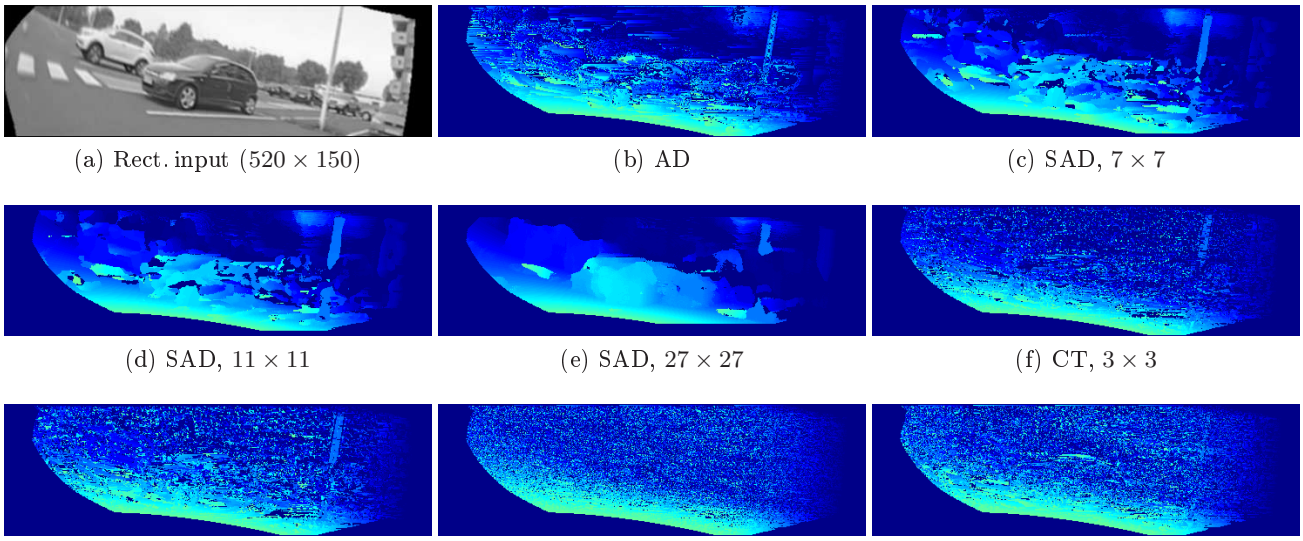
Depending on its distortion it is also possible that the image's resolution varies strongly with the view direction (*cf*. 6, Fig. 5). Thus, it is difficult to choose a useful angular resolution for the rectified image pair without over- or undersampling one image region or the other.

4. CAVEATS

The proposed computation of a rectified image pair is feasible. However, several properties raise a number of problems that do not influence the result so strongly in a classical stereo camera setup.

- The entire camera images may depict different parts of the scene and the overlapping field of view may be quite small. This can result in distinct differences of the recorded objects' contrast and brightness in the computed rectified image pair.
- The image resolution for the same view direction in both cameras can vary, resulting in different levels of sharpness and, thus, structuredness for a corresponding point in both rectified images.
- Depending on the accuracy of the distortion function (d in Sec. 3), especially at the image borders, and on the quality of the extrinsic calibration, corresponding image points may not lie in the same line, but in neighboring ones.
- Regarding the target setup, where the cameras are mounted at different positions on a vehicle the baseline is large compared to the distance of the scene objects to detect. This results in more and larger occlusions than in a classical setup.
- For the same reason intensity inconsistencies by non-Lambertian reflectance are more prominent.

In the following sections we show how these caveats can be addressed. While some will be avoided, others will be extenuated, but at cost of the final result's accuracy.



(a) Rect. input (520×150) (b) AD (c) SAD, 7×7

(d) SAD, 11×11 (e) SAD, 27×27 (f) CT, 3×3

(g) CT 5×5 (h) RD, 3×3 (i) RD, 11×11

Figure 3. Experiments with different dissimilarity measures. For each pixel the images show the disparity with minimal matching cost. A minimum disparity was computed by the ground plane constraint, but the cost for the proper ground disparity itself was not decreased in these examples.

5. PROCESSING PIPELINE

5.1 Dissimilarity measures

As dissimilarity measure we examined the following methods prominently conveyed in the literature:

- **Absolute Intensity Difference (AD)**: The cost is defined as the absolute difference of the grayvalues of the compared pixels.
- **Sum of Absolute Differences (SAD)**: The average of the absolute differences of a window of fixed size surrounding the compared pixels
- **Census-Transform (CT)**: Each pixel's neighborhood is coded as a bit-pattern flagging those neighbours that have higher intensity than the pixel itself. The Hamming distance of the bit-patterns defines the matching cost.
- **Rank-Difference (RD)**: The difference of the ranks of each pixel's intensity compared to the pixels in a surrounding window

Due to the differences in illumination of corresponding points (*cf*. Sec. 4) we apply a local brightness equalization for each pixel as a preprocessing for AD and SAD.

Furthermore, we made use of the knowledge of the ground plane with respect to the camera installation. By defining a minimum disparity for each pixel with help of the distance a point on the ground plane in the according direction would have. Additionally, in a post-processing step we reduced the matching cost for the ground plane disparity to encourage estimating the ground plane if no clear matching for the respective pixel had been found.

5.2 Accumulation

With real-time applicability in mind, we decide to deploy the well-established Semi-Global Matching (SGM) algorithm.[8] Explained briefly: To avoid optimizing a global cost function on the whole image the main idea is to minimize a one-dimensional cost function along several paths through every pixel. Traditionally, these paths

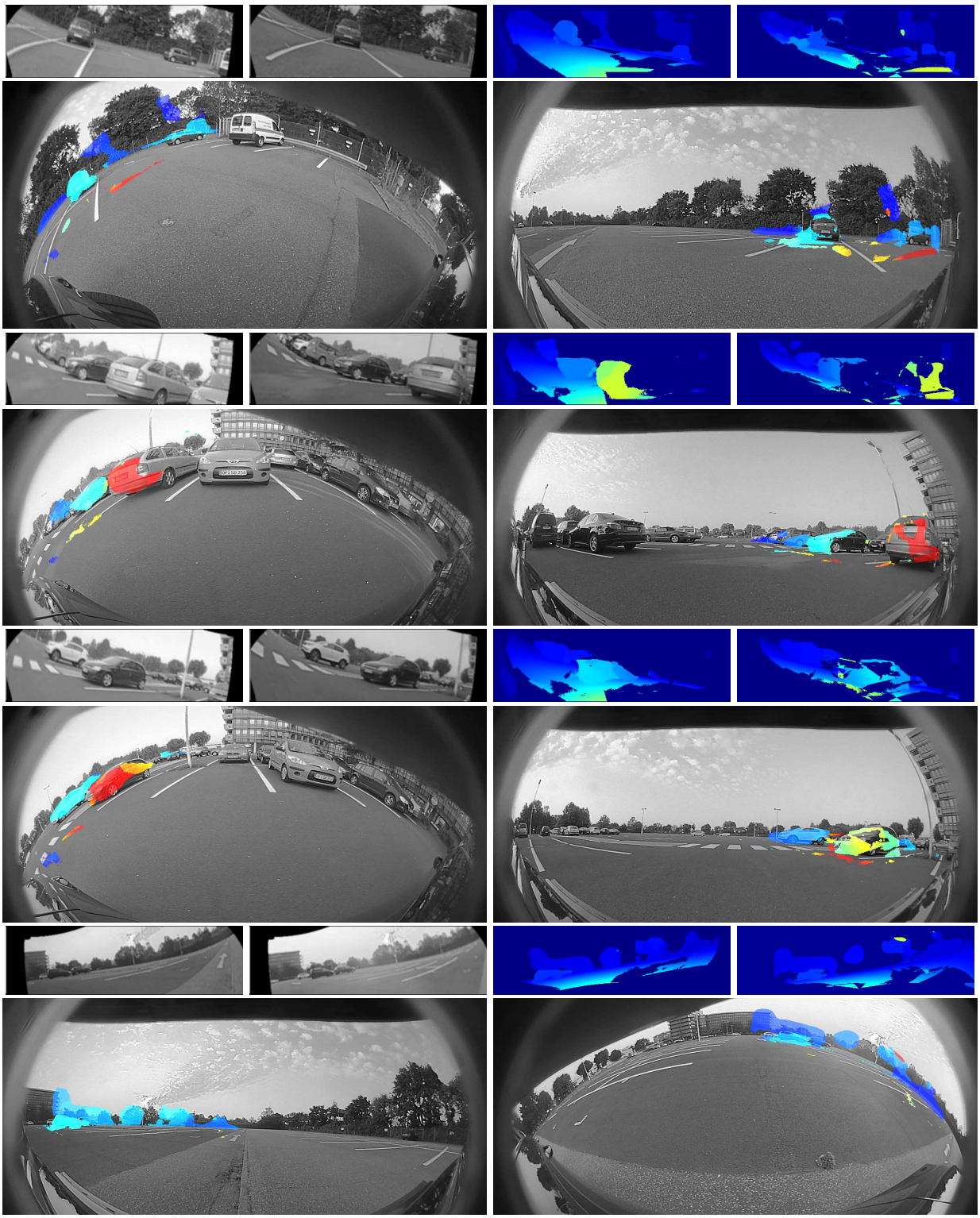


Figure 4. Some results from the complete proposed pipeline: correspondence analysis via a 23-by-23 patched SAD and accumulation using SGM. The left and right rectified image, both disparity images, and a visualization in the input camera images are shown (image points from the ground plane are not visualized). Pixels with a weak correspondence are neglected. The scenarios encompass static obstacles at varying distances with strong occlusions. Short sequences of this visualization can be found in the supplemental material (*cf*. Fig. 8 and Fig. 9).

Figure 5. The input and rectified images from the front (top) and right (bottom) camera of the presented topview setup. The overlapping field of view is indicated.

are horizontal, vertical and diagonal with respect to the image borders. The path cost function E for a disparity $D(p)$ assigned to all pixels p of the path is given by

$$E(D) = \sum_{p} \left(C(p, D_p) + \sum_{q \in N_p} P_1 \# \{|D_p - D_q| = 1\} + \sum_{q \in N_p} P_2 \# \{|D_p - D_q| > 1\} \right)$$

where N_p yields the neighboring pixels of p and $\# \{S\}$ is 1 if and only if S is true. P_1 and P_2 are parameters for penalizing disparity changes of 1 and > 1 respectively. Minimizing $E(D)$ yields a disparity for the path. The average of the minimal disparities of all paths traversing through p results in the final disparity.

Despite all effort the dissimilarity measure is unstable at times. We therefore choose P_2 significantly larger then P_1 to strongly favor a smooth disparity map with little outliers and discontinuities. Furthermore, we exclude pixels with a high-cost disparity as well in order to avoid outliers and account for occluded image regions which pose a significant problem (*cf*. Sec. 4). In Fig. 4 we depict results from three sequences using the front and either one of the two side-mirror cameras. The system works at approximately 0.25 frames per second, which shows the potential for real-time applicability after a thorough runtime optimization. In the current implementation no multi-processor or GPU functionality is made use of.

6. EXPERIMENTS

6.1 Camera Setup

We demonstrate the applicability of our approach using two similar vehicle-mounted topview systems with either four monochrome fisheye cameras with a view angle of 170°. They are mounted at the front and the rear trunk lid as well as at the two side mirrors and allow for a view fully surrounding the vehicles. The lens distortion function was determined by the method of Scaramuzza[11,12] using calibration patterns all over the camera image, thus, not restricted to the overlapping image regions. For reasons of brevity, in this paper, we exemplify the stereo procedure with the front and either one of the two side cameras. The baseline of the front and a side mirror camera is about 2.0 - 2.1 m with about 100° overlapping field of view.

Although an extrinsic calibration of the full system and individual lens distortion functions were known, it was necessary to perform a pair-wise calibration correcting the camera orientation by minimizing the quadratic distance of 25 to 80 point pairs from different scenes in the overlapping field of view of the respective camera pair. This additional effort mainly avoided the location of corresponding pixels in neighboring lines (*cf*. Sec. 4).

In order to avoid aliasing effects due to undersampling of the input images (*cf*. Sec. 3.2), we suggest to map the four neighbors of each pixel from the rectified image to the input image. The resulting quadrangle

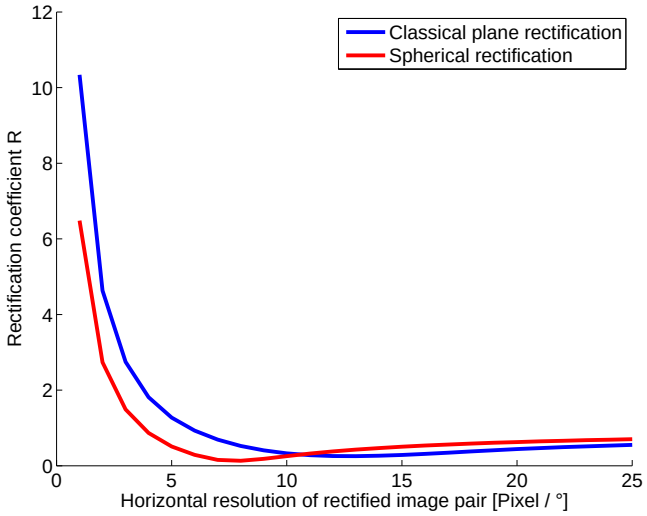


Figure 6. The aptitude of the two aforementioned rectification methods w.r.t. the angular resolution of the stereo image pair.

may span an image region of several pixels. Thus, we compute the average intensity over the four transformed neighbors' bounding rectangle which can be efficiently performed with the help of an integral image.[13] This method provides a compromise in computational cost and accuracy compared to integrating over the actual quadrangle and possibly interpolating the edge pixel gray values.

6.2 Rectification

In order to compare the proposed spherical rectification against the widely used planar rectification (*cf*. Sec. 3.2), which aims at undistorting the camera images to a common plane, we suggest to compute the following measure of aptitude.

In order to avoid over- or undersampling a good rectification should adapt to the characteristics of the camera images, i. e. it should use the given resolution as well as possible. Thus, one would expect that neighboring pixel positions in the rectified image pair belong to close pixels in the camera image. We, hence, remap the four neighbors $n_{i,1}, ..., n_{i,4}$ of each pixel p_i of the rectified image into the camera image yielding $r^{-1}(n_{i,1}), ...r^{-1}(n_{i,4})$ and compute the distance to the remapped pixel $r^{-1}(p_i)$ itself. A value larger than 1 suggests undersampling, a value less than 1 oversampling. Therefore, the average of the absolute deviation of the distances of the remapped neighbors from 1 yields a measure for the rectification procedure's aptitude R:

$$R = \sum_i \sum_{k=1}^{4} |1 - ||r^{-1}(n_{i,1}) - r^{-1}(p_i)||_2|$$

Figure 6 shows the course of R for varying image resolutions for both the spherical and the planar rectification. For low resolutions, spherical rectification clearly outperforms classical plane rectification. The sampling proposed in Sec. 3, thus, adapts well to the distortion of fisheye lenses and yields advantages in computation performance and memory requirements.

6.3 Entire Stereo System

Classical stereo systems are typically evaluated by two means: via a sequence with known depth information or via synthesized image data. We must rule out the former approach since adequate ground-truth data is not available on this exotic setup and generating it is currently unfeasible. The latter method, synthesizing image and ground truth data, is an option, the fisheye stereo setup's peculiarities are, however, mostly due to the influence of unknown or imprecisely-modeled distortion and extrinsic calibration. In a simulation approach, we

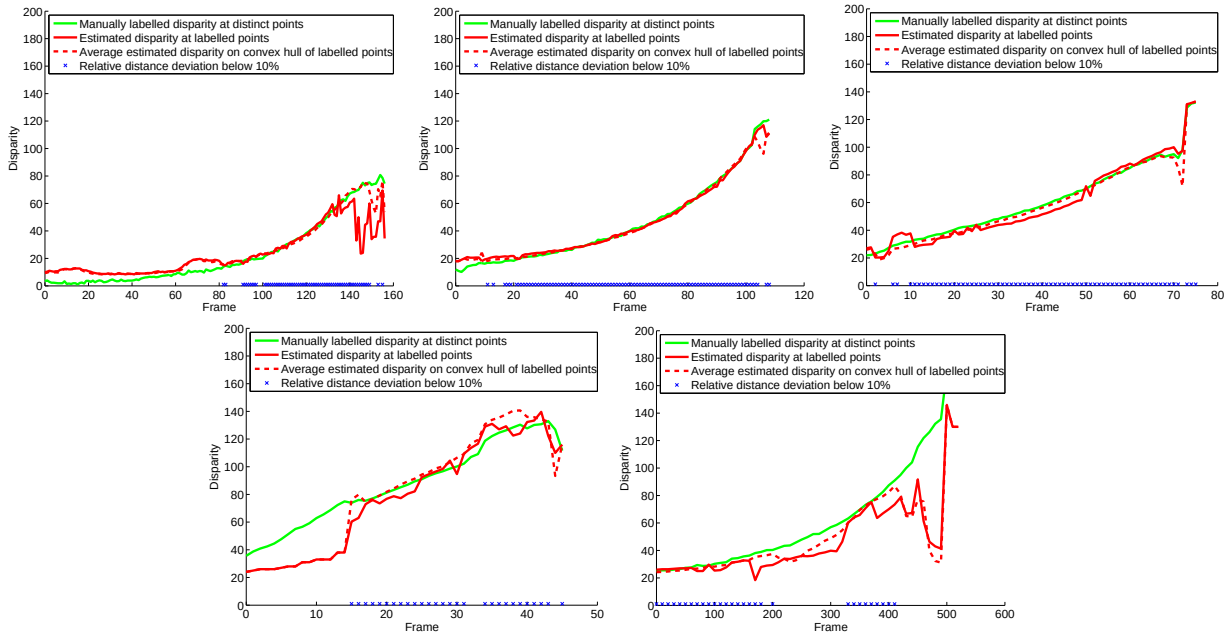


Figure 7. The course of the average disparity of the labeled points and the measured disparity measured at the labeled points' positions and averaged over their convex hull.

would not be able to account for this source of error correctly. In order to examine the characteristics and limits of the proposed stereo setup and processing pipeline in a lifelike situation, we, hence, perform a number of test maneuvers straightly approaching parked vehicles since these pose frequent safety-relevant obstacles. This is achieved by manually labeling and following 4 to 10 salient points of the vehicles in both camera images, *e.g.* edges of tires, lighting, and windows. At the same time the average disparity over the convex hull of all labeled points is recorded. Figure 7 shows the course of the disparity of the manually labeled and the disparity from the stereo pipeline for several test maneuvers. On average the first detection of the vehicle was at a distance of 16.1m $\pm$ 5.9m. On the other hand, the track was lost at an average proximity of 5.8m $\pm$ 1.6m due to the large disparity and strong perspective. Please note that both distances are given w.r.t. to the side mirror camera. While approaching the obstacle, the stereo system was able to detect it in 93% $\pm$ 12% of the frames. The average positioning error was measured at 0.42m $\pm$ 0.38m which is equivalent to a relative error of 3.7% $\pm$ 2.6%.

7. DISCUSSION AND CONCLUSIONS

We state that the choice of the dissimilarity measure (*cf*. Fig. 3) is the most crucial in the processing pipeline. Well-established correspondence methods like CT or RD perform very poorly on the rectified images. This can be attributed to the strong perspective distortions as well as the changing resolution in different parts of the image. Our solution to circumvent these problems was to use rather large patches, thus, introducing a smooth correspondence very early in the process.

Regarding the SGM refinement the chosen parameters $P_1 = 15$ and $P_2 = 120$ again point at a strong smoothness regularization of the problem since the ratio of both penalty parameters is large. Hence, it encourages gradual disparity slopes instead of discontinuities.

To summarize, in order to deal with occlusions, perspective distortions, and resolution changes (*cf*. Sec. 4) we have to find a useful trade-off between smoothness, spatial resolution, and computation cost. In contrast to *classic* stereo approaches, this compromise clearly has to be drawn towards smoothness.

In our setup wide-angle topview cameras, albeit not designed for this purpose, were taken advantage of as a stereo distance sensor within their overlapping field of view. We obtained an additional general purpose distance sensor with a range of 5 to 15 meters. Thus, areas of application are near-field obstacle detection and collision

warning with low relative velocity. Furthermore, stereo information can often be made of good use as a pre-segmentation for other image processing problems like object recognition.[14] Hence, the presented approach can also facilitate other tasks performed on the topview system. We, therefore, see the topview stereo approach as a supplemental sensor for a variety of driver assistance tasks.

We consider the most potential for further development in a refinement of the correspondence measure. In the future, we will look into ways of handling strong perspective distortions for a more accurate matching. Furthermore, we want to evaluate the system's capability in several scenarios, *e.g.* in backwards maneuvering situations or when changing lanes during highway scenarios.

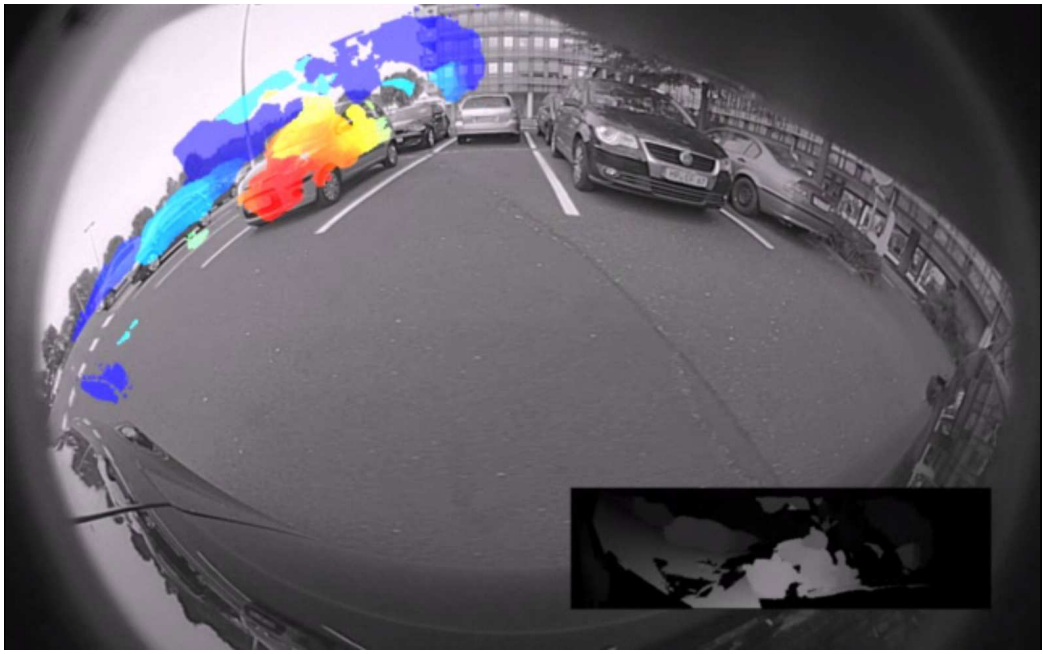


Figure 8. Video 1: The color-coded disparity estimation in a sequence seen from the right side mirror camera. The vehicle approaches a line of parked cars. http://dx.doi.org/doi.number.goes.here

Figure 9. Video 2: The color-coded disparity estimation in a sequence seen from the right side mirror camera. The vehicle performs a short drive through a roofed parking deck. http://dx.doi.org/doi.number.goes.here

REFERENCES

[1] Houben, S., Komar, M., Hohm, A., Lüke, S., Neuhausen, M., and Schlipsing, M., "On-vehicle video-based parking lot recognition with fisheye optics," in [*Proceedings of the IEEE Annual Conference on Intelligent Transportation Systems*], 7 – 12 (2013).

[2] Hartley, R. I. and Zisserman, A., [*Multiple View Geometry in Computer Vision*], Cambridge University Press, second ed. (2004).

[3] Pollefeys, M., Koch, R., and Gool, L. J. V., "A simple and efficient rectification method for general motion.," in [*Proceedings of the International Conference on Computer Vision*], 496–501 (1999).

[4] Abraham, S. and Förstner, W., "Fish-eye-stereo calibration and epipolar rectification," *Journal of Photogrammetry and Remote Sensing* **59**(5), 278–288 (2005).

[5] Hirschmüller, H. and Scharstein, D., "Evaluation of cost functions for stereo matching," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 1–8 (2007).

[6] Hirschmüller, H. and Scharstein, D., "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Transactions on Pattern Analalysis and Machine Intelligence* **31**(9), 1582–1599 (2009).

[7] Forstmann, S., Kanou, Y., Ohya, J., Thuering, S., and Schmitt, A., "Real-time stereo by using dynamic programming," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop Volume 3*], 29 (2004).

[8] Hirschmüller, H., "Accurate and efficient stereo processing by semi-global matching and mutual information," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 807–814 (2005).

[9] Hirschmüller, H., "Stereo vision in structured environments by consistent semi-global matching," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 2386–2393 (2006).

[10] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R., "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research* (2013).

[11] Scaramuzza, D., Martinelli, A., and Siegwart, R., "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in [*Proceedings of IEEE International Conference on Computer Vision Systems*], 45 (2006).

[12] Scaramuzza, D., Martinelli, A., and Siegwart, R., "A toolbox for easy calibrating omnidirectional cameras," in [*Proceedings of IEEE International Conference on Intelligent Robots and Systems*], 7–15 (2006).

[13] Viola, P. and Jones, M., "Rapid object detection using a boosted cascade of simple features," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 511–518 (2001).

[14] Keller, C., Enzweiler, M., Rohrbach, M., Fernandez Llorca, D., Schnorr, C., and Gavrila, D., "The benefits of dense stereo for pedestrian detection," *IEEE Transactions on Intelligent Transportation Systems* **12**(4), 1096–1106 (2011).