Surrounding View for Enhancing Safety on Vehicles

J. Javier Yebes, Pablo F. Alcantarilla, Luis M. Bergasa, Á. González and J. Almazán

***Abstract*— This paper presents a vision system based on several wide angle cameras and robust methods to compose a view of the vehicle perimeter in only one image. Our proposal can be split into two stages: firstly, an offline calibration step and secondly, an online panorama composition given the acquired images from the cameras. The images are horizontally undistorted, warped and stitched together according to lookup tables (LUTs) generated during the calibration process. The offline part of the method calibrates every camera individually and computes the pixel mapping required for the images transformation into the surrounding view. This mapping encodes a special undistortion process and the image registration between adjacent cameras. In particular, the proposed method does not fully undistort the images but it projects them onto a cylindrical surface to reduce the amount of radial distortion and facilitate the panoramic stitching of the images. Our approach includes a feathering algorithm for seams removal and a polar conversion of the panoramic view. This method has been successfully tested on a small wheeled robot and it is fully scalable to larger platforms such as an armored patrol. Vehicles safety can be enhanced thanks to perimeter monitoring.**

I. Introduction

Special vehicles such as armored patrols are usually equipped with tiny windows not wide enough to watch the perimeter of the vehicle from the inside. However, they can be benefited from a vision system composed of several cameras mounted on the outer chassis of the vehicle. Then, a panoramic view of the vehicle surroundings can be generated from the cameras and displayed on onboard screens. In fact, onboard officers can monitor the panoramic and individual images from the inside and/or transmit to other neighbouring patrols. Then, this system can be useful for big commercial vehicles such as trucks, vans or land-rovers.

In this work, we propose a vision system based on several wide angle cameras and robust methods to compose a view of the vehicle perimeter in only one image. Panoramic views are characterized by very large horizontal Fields of View (FOVs) and they typically are used to generate 360° views of nature or city scenes. Wide angle cameras provide images with large FOVs, which makes them ideal candidates for generating views of vehicle surroundings. However, the generation of a panoramic view from wide angle images is not a simple matter of correcting them into perspective images [1]. In fact, undistortion is useless to create panoramic views because the edges of the corrected images are stretched out as a result of the perspective mapping. Besides that, their FOV is small by definition. Consequently, we propose a different approach through the use of a cylindrical surface to correct the distortion only on the horizontal axis.

On one hand, existing comercial systems [2] [3] are based on the use of several cameras (more than four) with narrow FOVs (around 50°), which are attached together in a small cylindrical or spherical device. Then, the stitching of the adjacent images is easy to perform because the cameras are close to each other in the same device. Besides, these systems need to be installed on top of the vechicle's roof at some height in order to capture proper images of the surroundings.

On the other hand, our approach proposes a more natural integration of the cameras on the car chassis, using 4 or 6 six wide angle lenses separated from each other. Considering this deployment, the generation of a 360° view and the image stitching are complex tasks, which are solved by the methodology proposed in this paper.

Our approach also facilitates the implementation of the online paranorama composition on a Field-Programmable Gate Array (FPGA), which can be easily embedded on intelligent vehicles. In addition, our method is also opened to different Human Machines Interfaces (HMI) for image displaying, as the one presented in [4]. This work briefly describes a technology known as "Multi Angle View (MAV)", which provides a wraparound view of the vehicle surroundings. They use four cameras around the vehicle to create a 360° view. Their HMI displays the vehicle surroundings from different viewpoints and they claim that the system helps the driver in different driving situations such as parking, turning and watching blind spots. The final image representation is in the form of a bird's eye view perspective, which causes high image distortion for the objects around the vehicle.

The work by Liu et al. [5] proposes a bird's eye view approach too, but using six cameras. They rely on an elaborated method based on Wendland functions to seamlessly align and stitch the images. However, they undistort the images from the vehicle-mounted wide angle cameras, so that the stitched images have a FOV similar to perspective ones. Hence, they disregard wide

J. Javier Yebes, Pablo F. Alcantarilla, Luis M. Bergasa, Á. González and J. Almazán are with Department of Electronics, University of Alcalá. Alcalá de Henares, Madrid, Spain. e-mail: javier.yebes, pablo.alcantarilla, bergasa, alvaro.g.arroyo, javier.almazan@depeca.uah.es

angle lenses properties requiring the use of two more cameras to complete the vehicle surrounding view.

There are some other works on truck drivers assistance to increase safety while driving large vehicles. The authors of [6] propose the installation of four catadioptric cameras to remove blind spots and obtain a bird's eye view of the truck perimeter. Moreover, they include the approximate path of motion over the same resultant view.

In our work, we propose an alternative method to compose a surrounding view of the vehicle perimeter using a vision system similar to the references cited before. We are aware of the private technologies presented in [4] and [5], but our approach is able to deal with some of the inconvenients of those systems. For instance, our panoramic view does not suffer from bird's eye view distortion on the closer areas of the vehicle. In addition, we propose the use of cheap and small cameras compared to the catadioptric ones used in [6].

In the remainder of the paper, we introduce the system and the proposed method in section II. Sections III and IV describe in detail the algorithms for the offline system calibration and the online panorama composition. We validate our approach using 4 wide angle cameras mounted on a wheeled robot in section V. Finally, we remark the main conclusions and future works guidelines in Section VI.

II. System Overview and Proposed Method

This work proposes a system that integrates 4 or 6 cameras (depending on vehicle dimensions) mounted in the outer chassis of the vehicle. See Fig. 1.

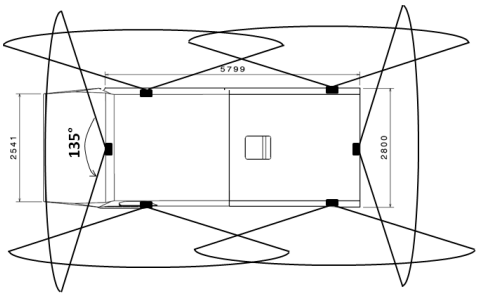


Fig. 1. Top view of an armored patrol that ilustrates the placement of 6 cameras and their approximate field of view.

The system is designed to have a full view of the perimeter and supplement the vision of blind spots, which is a critical monitoring task for armored patrols. Every camera is equipped with a wide angle lens and this set provides a field of view (FOV) of 135°. A 360° view of the vehicle surroundings can be composed in only one image and displayed from a desired viewpoint. Besides, the images can be also individually displayed in the screens onboard. We propose a method to stitch the adjacent images together in order to obtain the final panorama composition. Firstly, an offline process performs the system calibration and computes the lookup tables (LUTs) required for the online stage, which captures images in real-time and composes the 360° view of the vehicle surroundings. The offline computed LUTs can be stored in memory and used by FPGA or other small electronic processing systems, which can be easily integrated in a vehicle.

Fig. 2 depicts the main steps of the proposed method. During the offline calibration, the intrinsic parameters of the cameras are obtained. Then, the images are projected onto a cylindrical surface to remove radial distortion only for the horizontal axis. The final goal is to stitch images to have a panoramic view, so we can disregard the distortion in the vertical axis, focusing only in the horizontal one. These undistorted images are used to perform the registering step to find the transformation that relates adjacent images. Finally, the LUTs are computed and stored.

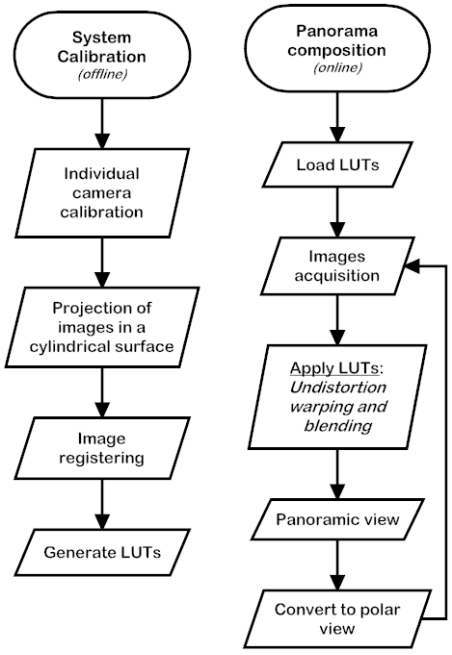


Fig. 2. Flowchart of the proposed method, which is divided into an offline process for system calibration and the online one that composes the final representation of the vehicle surroundings.

On the other hand, the online process applies these pixel mapping tables over the set of online captured images. Indeed, the techniques encoded by the LUTs involves image horizontal undistortion and warping. As a result, two panorama representations can be displayed: a rectangular (see Fig. 3) and a circular view.

Once the images are stitched together, some undesired artifacts can appear in the overlapping areas between adjacent images. This is solved in real time through the use of blending techniques. The following two sections describe in detail the algorithm within the two stages of our method.

Fig. 3. Sample rectangular panorama of the vehicle surroundings in a parking.

III. System Calibration

The key parts within the calibration are the projection of distorted images into a cylindrical surface and the image registration. Before that, an invididual camera calibration process [7] [8] allows to determine the focal length for every camera, which is required in order to perform the projection.

A. Image projection onto a cylindrical surface

Fisheye lenses provide a field of view near 180°, which is well suited for the removal of blind spots and the composition of a 360° surrounding view. This property and its cost-effectiveness, they both facilitate the installation of a fewer number of cameras while providing overlapping image regions between adjacent cameras. Particularly, the second fact is strictly mandatory in order to get the final panoramic view. We have empirically determined the requirement of about 45° of overlapping between the fields of view of adjacent cameras, which affects their placement around the vehicle chassis. This overlappping area allows the matching between features in the adjacent images and the computation of their transformation matrix.

On the other hand, fisheye lenses cause radial distortion in the captured images. They project the frontal hemispheric scene onto the flat surface of the camera CCD. As a consequence, the stitching of several wide angle images is non-trivial and difficult to solve. Besides that, there are two additional factors to be considered regarding the reduction of the camera FOV, which is affected by the size of the CCD, e.g. reduction up to 50° for the case 1/4" CCD. Secondly, image undistortion is a transformation process that converts the original captured image into a perspective one with a smaller field of view (FOV). There are some reference works on this topic [8] [9], but they have been proved useless for creating panoramic images. The undistorted images are stretched out in the edges, being the effect greater as one moves further from the principal axis. Hence, some approximations have to be considered to make the problem feasible. More specifically, the proposed method assumes that the objects in the scene are sufficiently far away from the camera, so that we can ignore stereo disparity effects. We propose a correction motivated by the large horizontal dimension of a panoramic view. The vertical FOV does not need to be increased, but the opposite does happen while stitching images from adjacent cameras. Thus, the undistortion process relies only on the horizontal axis and it is achieved considering a cylindrical surface model as depicted in Fig. 4.

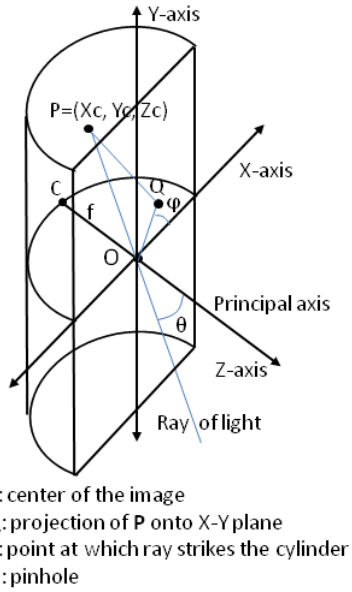


Fig. 4. Compositing surface to project the radially distorted image onto a cylinder to correct distortion only in the horizontal axis.

The distorted image is projected onto a compositing surface of a cylinder considering the pinhole camera placed on the center of the cylinder and its principal axis along z axis. The radius is equal to the focal length. Hence, the point P in Fig. 4 represents a scene point in the cylinder surface and the point Q is the projection onto the image plane. Unrolling the image formed on the cylinder and stitching to the adjacent images results in the final panoramic representation.

The image correction is performed according to [10] and [11] and the equations (1) and (2), which define a linear projection model for the fisheye camera.

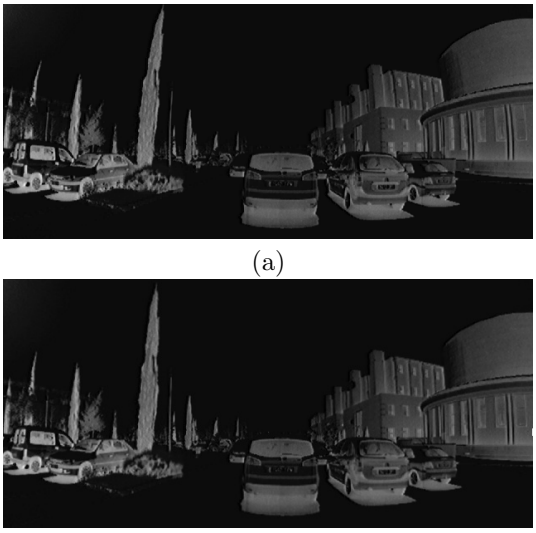
$$x_f = f \cdot \theta \cdot cos\varphi = f \cdot \lambda \cdot sin(\frac{x_q}{f}) \qquad (1)$$

$$y_f = f \cdot \theta \cdot sin\varphi = \lambda \cdot y_q \tag{2}$$

$$\lambda = \rho^{-1}\left[tan^{-1}\left\{\frac{\rho}{cos(\frac{x_q}{f})}\right\}\right] \tag{3}$$

$$\rho = \sqrt{(\frac{y_q}{f})^2 + sin^2(\frac{x_q}{f})} \tag{4}$$

Equations (1) and (2) establish the relationship between every pixel $I_u(x_q, y_q)$ in the unrolled image and in the captured image $I_c(x_f, y_f)$, which is distorted due to the fisheye lens. The angles θ and φ are obtained from the projection of point P into Q as represented in Fig. 4. On the other hand, Fig. 5 shows a sample of the original and corrected images. The images have been cropped and inverted to perceive the correction in horizontal axis. The differences between the images can be appreciated comparing the tree and the building in both images.



(a)

(b)

Fig. 5. Sample images. a) Original distorted image captured by the wide angle camera and b) corrected version.

B. Image registering

Stitching the images requires their registration to determine the overlapping region. This means estimating the transformation matrices that relates the images from adjacent cameras. Considering that our system is not pure rotational and the appropiate image stitching techniques [10], we propose to use a planar perspective motion model after the cylindrical surface projection of the images. The cameras are not mounted together on the same vertical axis and this implies a translation and rotation between their camera coordinates systems. Indeed, camera alignment errors are also considered. Additionally, not all the objects in the scene are very far away, so that the results using a pure rotational model yield incorrect panoramic representations. However, our proposed model requires a placement of the cameras so that, all their principal axes lie in the same horizontal plane.

The planar perspective motion model that includes image rotation and translation can be described using a 3x3 affine matrix as shown in (5).

$$\tilde{x_2} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{pmatrix} \cdot \tilde{x_1} = H_{21} \cdot \tilde{x_1} \tag{5}$$

where $\tilde{x_1} = (u_1\ u_1\ 1)^T$ and $\tilde{x_2} = (u_2\ u_2\ 1)^T$ are the pixels on each adjacent image respectively. In order to estimate the 6 unknown parameters of H_{21} we first have to match a set of points in both images. The simplest way of doing it is by manually selecting a minimun of 3 points. However, choosing 12 corresponding points in both pictures from adjacent cameras yields better results, as we have empirically tested. We are able to find a robust set of inliers using RANSAC [12] and compute the affine matrix as the solution of a least squares fitting problem given the pairs of point correspondences between the images. This is an offline task which requires a planar reference object in the overlapping region of both images. We use a common calibration rig [9] and select 12 matching points between the two adjacent images.

Reliable feature matching techniques have been also successfully tested in this work, instead of manually selection of features. SURF detector and invariant descriptors [13] have been used to find and match visual features between images from adjacent cameras. However, they do not improve the accuracy of the affine matrix estimation, so we have decided to keep a simpler version of manual selection of matching points. In addition, the alignment errors between the images were not reduced either. Consequently, we propose a global bundle adjustment [14] in order to readjust the parameters of all the transformation matrices at the same time. This motion model refinement is based on the Levenberg-Marquardt algorithm [15]. The alignment error is mainly due to a drift in the y pixel coordinate after applying every affine transformation to adjacent images. Hence, we set as a constraint that the drift equals zero, which means that the first and last images of the panoramic view are forced to match in the vertical coordinate. This method is robust for small vertical drift corrections. However, there is a main constraint in the cameras motion model, as it was stated before: their principal axes have to be in the same horizontal plane.

Finally, the LUTs that transform every image into the final panoramic representation can be computed according to the projection and affine transformation described above.

IV. Online Panoramic View Composition

The online stage of the proposed method is in charge of capturing images and warping them into the final representation. During system calibration, all the LUTs required for the mapping of every image into the final

panoramic view were computed. However, there is still an important step until the final image representation. Images acquired from different cameras are prone to exposure differences, which cause visible seams on the stitched image. We propose to use a feathering technique [10] for image blending in order to remove this undesired effect. However, feathering does not correct blurring (due to mis-registration) or ghosting (due to moving objects). In the first case, we have observed very low blur, which can be ignored. On the other hand, we assume some ghosting effects due to the relative movement between objects and the vehicle.

A. Image blending

Blending deals with the pixels in the overlapping area between adjacent images and it seamlessly stitches the images into a continuous panorama. The gray level of the pixels in the overlapping region is computed as a weighted-average, whose value depends on the distance to the edge of this area on each image. In particular, we propose a feathering technique that loops over each row in the warped image and recomputes the pixel values according to (6).

$$I_w(x_i, y_j) = \frac{d_i}{l} \cdot I_1(x_i, y_j) + (1 - \frac{d_i}{l}) \cdot I_2(x_i, y_j) \quad (6)$$

Both images are referenced to the same coordinate sytem, which is placed on the final warped and blent result. The algorithm detects the overlapping line on each row, computes its length l and perfoms the weighted-average indicated in (6). The parameter d_i represents the distance from the starting overlapping pixel to the current pixel x_i. Basically, the algorithm calculates the mean value at the center of the overlapping region and penalizes one of the images, as it gets closer to the boundaries of the region.

B. Circular view representation

Once the image blending process is finished, the final panoramic view (Fig. 3) can be displayed on a screen onboard the vehicle. However, this image represents a field of view of 360° around the vehicle. Thus, its horizontal dimension is too wide. We propose a more practical and appropiate representation for displaying the image on non-panoramic onboard screens. An additional mapping of pixels can be computed to generate a circular view (Fig. 8) of the vehicle surroundings considering the previously generated rectangular panorama. This mapping is stored as another LUT for speeding up the warping process. We perfom a cartesian-to-polar coordinates conversion according to (7) and (8).

$$r = \sqrt{(x^2 + y^2)} \quad (7)$$

$$\theta = atan(\frac{y}{x}) \quad (8)$$

In addition, a bilinear interpolation is carried out to complete the image conversion.

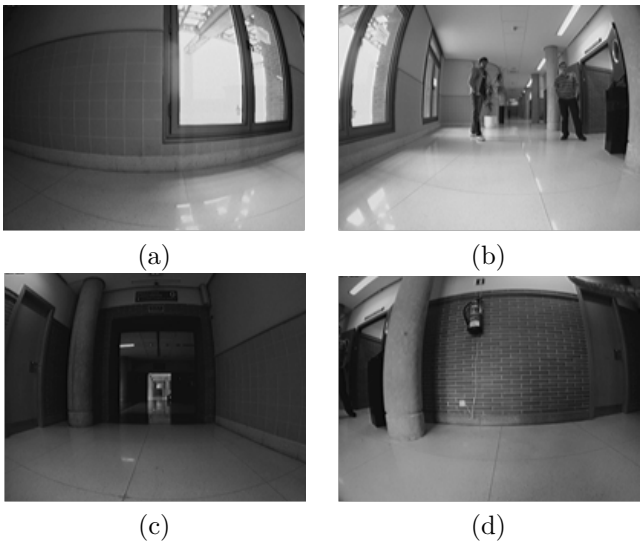
V. Validation on a Wheeled Robot

The methodology introduced in section II has been tested on a Pioneer 3-AT robot. These experimental tests have not been carried out in real armored patrols yet, but we plan to do it in the next future. The proposed system has been designed fully scalable. The offline calibration computes the required LUTs for the number of cameras mounted on the vehicle. The online process for panorama composition and the LUTs can be easily integrated in embedded systems.

Four cameras have been orthogonally placed at the same height in a metallic structure on the robot, as it is shown in Fig. 6. They are Unibrain Fire-i digital cameras [16] with a Sunex DSL209 lens [17], which provides a FOV of 123°. The resulting overlapping area between adjacent cameras is wide enough given the dimensions of the robot.

Fig. 6. Autonomous robot Pioneer 3-AT with 4 cameras mounted on a metallic structure.

The robot was programmed to drive autonomously through indoor corridors. Fig. 7 displays 4 samples images from each of the mounted cameras.



(a) (b) (c) (d)

Fig. 7. Images from every camera installed on the robot. The orientation is as follows: a) left, b) front, c) back and d) right.

The image acquisition process was performed using a laptop with a 2GHz Intel core and 2GB of memory, running Kubuntu 10.04 and OpenCV library [18]. The

online panorama composition runs at a rate of 5 frames per second (fps) given that the cameras synchronization, image acquisition and processing are done by software. This process can be speeded up using a quad video board controlled by a FPGA. Besides, the LUTs computed during system calibration can be stored in a flash memory, so that the warping and blending steps can be entirely implemented on the FPGA.

The final representation is displayed in Fig. 8 as a circular view of the robot surroundings.

Fig. 8. Circular view representation of the robot surroundings after warping and blending of the 4 images.

VI. Conclusions and Future Work

This paper has presented an alternative method for surrounding view composition given several images captured by wide angle cameras mounted on a vehicle. The proposed system is fully scalable and we have conducted experimental tests on a small wheeled robot using 4 cameras. As a result, two representations in the form of 360° FOV images (panoramic and circular) can be computed in real-time and easily integrated on embedded systems for intelligent vehicles.

In our approach we tackle exposure effects due to different camera views, but there is still work to do to reduce other undesired effects such as ghosting and blurring.

On the other hand, our proposal is based on the assumption of the coplanar placement of the cameras. As future works, we also plan to do research on camera autocalibration and the use of different compositing surfaces in order to relax that constraint and achieve better panoramic representations.

The system deployment on an armored patrol and other large vehicles is being under research and we hope to have further experiments in near future. Besides that, a Human Machine Interface to present the panoramic and/or individual images from the cameras is also a desirable feature to include in the next upgrades of the system.

VII. Acknowledgments

This work was supported in part by the Spanish Ministry of Education and Science (MEC) under grant TRA2011-29001-C04-01 (ADD-Gaze Project), by the multinational corporation FICOSA INTERNATIONAL and by Comunidad de Madrid through the project RoboCity2030 II-CM (CAM - S2009/DPI-1559).

References

[1] Altera, "White paper: A flexible architecture for fisheye correction in automotive rear-view cameras." 2008, http://www.altera.com/literature/wp/wp-01073-flexible-architecture-fisheye-correction-automotive-rear-view-cameras.pdf.

[2] "Point Grey Ladybug3 360° Video Camera." [Online]. Available: http://www.ptgrey.com/products/ladybug3/ladybug3_360_video_camera.asp

[3] "Immersive Media, Dodeca 2360 Camera System." [Online]. Available: http://immersivemedia.com/products/capture.shtml

[4] S.Taniguchi and M. Yamada and T.Hayashida and F.Bandou and J.Kawai and K.Yano and T.Gomi and K.Kataoka and S.Shimizu and H.Yamada, "Multi Angle Vision System to Supplement Driver's Visual Field," 2010, (IT ITS)Auto Bus Eng System Engineering Dept, Fujitsu Ten Co. Ltd.

[5] Y.-C. Liu, K.-Y. Lin, and Y.-S. Chen, "Bird's-eye view vision system for vehicle surrounding monitoring," in *Robot Vision*, ser. Lecture Notes in Computer Science, G. Sommer and R. Klette, Eds. Springer Berlin / Heidelberg, 2008, vol. 4931, pp. 207–218.

[6] T. Ehlgen, T. Pajdla, and D. Ammon, "Eliminating blind spots for assisted driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 4, pp. 657–665, 2008.

[7] "Documentation: Camera Calibration Toolbox for Matlab," 2007, http://www.vision.caltech.edu/bouguetj/calib_doc.

[8] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, ser. CVPR '97, Washington, DC, USA, 1997, pp. 1106–1112.

[9] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 1998.

[10] R. Szeliski, *Computer Vision: Algorithms and Applications*. New York: Springer, 2010.

[11] D. Salomon, *Transformations and Projections in Computer Graphics*. Springer, 2006.

[12] R. Bolles and M. Fischler, "A RANSAC-based approach to model fitting and its application to finding cylinders in range data," in *Intl. Joint Conf. on AI (IJCAI)*, Vancouver, Canada, 1981, pp. 637–643.

[13] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[14] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. LNCS, W. Triggs, A. Zisserman, and R. Szeliski, Eds. Springer Verlag, Sep 1999, pp. 298–375.

[15] M. Lourakis, "levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++," [web page] `http://www.ics.forth.gr/~lourakis/levmar/`, 2004.

[16] Unibrain, "Unibrain fire-i dc specifications," http://www.unibrain.com/Products/VisionImg/Fire_i_DC.htm, 2009.

[17] S. DSL, "Sunex fisheye dsl lenses," http://www.optics-online.com/, 2009.

[18] "OpenCV - Open Source Computer Vision." [Online]. Available: http://opencv.willowgarage.com/wiki/