Data fusion of multi-spectral cameras on a low-power processing platform for self-sufficient outdoor operation

Andreas Reichel*[a], Nico Peter[a], Jens Döge[a], Holger Priwitzer[a], André Kasper[b], Mike Ludwig[c], and Bernd Ziems[d]

[a]Fraunhofer Institute of Integrated Circuits IIS, Zeunerstraße 38, Dresden, Germany
[b]AT - Automation Technology GmbH, Hermann-Bössow-Straße 6, Bad Oldesloe, Germany
[c]dresden elektronik ingenieurtechnik gmbh, Enno-Heidebroek-Straße 12, Dresden, Germany
[d]Minimax Viking Research & Development GmbH, Industriestraße 10/12, Bad Oldesloe, Germany

ABSTRACT

Multi-spectral camera set-ups may generally allow for creating surveillance applications even under unfavorable conditions, such as low-light environments or scenes involving vastly different lighting conditions. A high-resolution color camera, a high-dynamic-range camera and an infrared thermal camera were combined into a self-sufficient platform for continuous outdoor operation. The sheer amount of produced data poses a serious challenge, both in terms of available bandwidth and processing power, because self-sufficiency requires using relatively low-power components, and privacy, as high-resolution, multi-spectral image data are sensitive information. Thus, relevant objects of interest had to be efficiently extracted, tracked and georeferenced on the sensor platform. These data, from one or more sensorheads, are then sent via WLAN or mobile data link to a central control unit, possibly anonymized, e.g. prompting immediate action by a human operator in a disaster response use case, or stored for further offline analysis when used in the framework of "Smart City". Applying the classic stereo vision approach would require calibrating both intrinsic and extrinsic parameters of all cameras. The input data's multi-spectral nature complicates the correspondence problem for extrinsic parameter calibration and subsequent stereo matching, while intrinsic parameter calibration according to the pinhole camera model is made difficult due to the cameras having to be focused at infinity. However, by making certain reasonable assumptions about the observed scene in typical use cases, accepting a possible loss in localization accuracy, camera calibration could be limited to the bare minimum and less computational power was required at run-time.

Keywords: multi-spectral image processing, foreground detection, image stabilization, object tracking, image fusion, privacy by design, video surveillance

1. INTRODUCTION

Cameras with different spectra and dynamic range characteristic can in principle be combined for automated surveillance applications capable of operating even under unfavorable conditions, such as low-light environments or scenes involving vastly different lighting conditions. Such a platform, designed for continuous outdoor operation, may e. g. aid in disaster response scenarios by assisting a human operator in distinguishing humans from debris, possibly prompting immediate action, or may be used in crowded areas to measure people flows. Such a system should, very generally, perform some kind of foreground segmentation and output only segmented, possibly classified "objects", which may directly refer to distinct physical objects or abstract properties such as optical flow. Self-sufficiency is required, because continuous power supply cannot always be guaranteed, i. e. only relatively low-power processing components can be used. Cameras of different resolution, spectrum and dynamic range characteristic should be combinable for different scenarios. Thus, for demonstration purposes, a 6-megapixel color camera, a high-dynamic-range camera based on the 1-megapixel Vision-System-on-Chip[1] (VSoC) and an Automation Technology IR thermal camera have been combined into a single self-sufficient sensor node for outdoor operation. The goal is to create an efficient, portable and reusable toolkit, which can easily be adapted to different (multi-spectral) surveillance use cases.

* andreas.reichel@eas.iis.fraunhofer.de; phone 49 351 4640-733; eas.iis.fraunhofer.de

Photonics and Education in Measurement Science 2019, edited by Maik Rosenberger, Paul-Gerald Dittrich, Bernhard Zagar, Proc. of SPIE Vol. 11144, 111440Y
 · doi: 10.1117/12.2531351

Figure 1. The demonstration sensorhead hardware.

2. CHALLENGES

In general, given a calibrated system of multiple cameras, the classic stereo vision approach can be applied to precisely locate objects in 3-D space. However, feature detection and description, which is needed for camera registration and stereo matching, is complicated by the input data's multispectral nature. Typical state-of-the-art keypoint feature detectors and descriptors such as ORB[2] (oriented FAST and rotated BRIEF) and SIFT[3] (scale-invariant feature transform) generally rely on local areas of gradient magnitudes, angles or other grayvalue-derived image features, which are not spectral-invariant and may thus appear differently and possibly completely unrecognizable across different spectra. Although a spectral-invariant extension of SIFT[4] has been shown to produce high detection and matching accuracy for hyperspectral images of different cameras, its computational complexity, runtime and reliance on three-dimensional hyperspectral image sequences make it unsuitable for real-time applications under low-power conditions in general and this use case in particular. Furthermore, it has only been applied to images in the visible light and near-infrared spectrum up to 1000 nm, not in the thermal infrared range. Due to this lack of a general, spectral-invariant keypoint detector and descriptor, any multi-spectral stereo vision set-up, if at all possible, has to be tailor-made for the specific camera set-up.

The data volume of such a set-up naturally increases linearly with the number of installed cameras, posing a challenge in terms of available bandwidth and processing power. This becomes especially problematic, if raw image data must not be transmitted from the sensor node, e. g. due to privacy concerns which are inherent to the nature of the acquired image data. In this case, image processing must be performed on the sensor node, which is limited by the need for low-power processing components. Depth estimation from stereo image pairs using epipolar geometry is not computationally expensive, but would require a reliable spectral-invariant feature detector and descriptor. Various methods for depth estimation from single images have been proposed,[5,6] but none of them are computationally inexpensive enough for use in the desired set-up. The VSoC camera can help mitigate this issue by operating as a software-defined smart camera, allowing for many localized, parallelizable image processing operations to be performed directly on the image sensor.[7] Data reduction as early as possible can reduce both the required bandwidth and the computational strain further along the processing chain. This unique VSoC capability was employed in the form of entirely on-chip background segmentation in related work, based on a previously developed presence detection approach.[8]

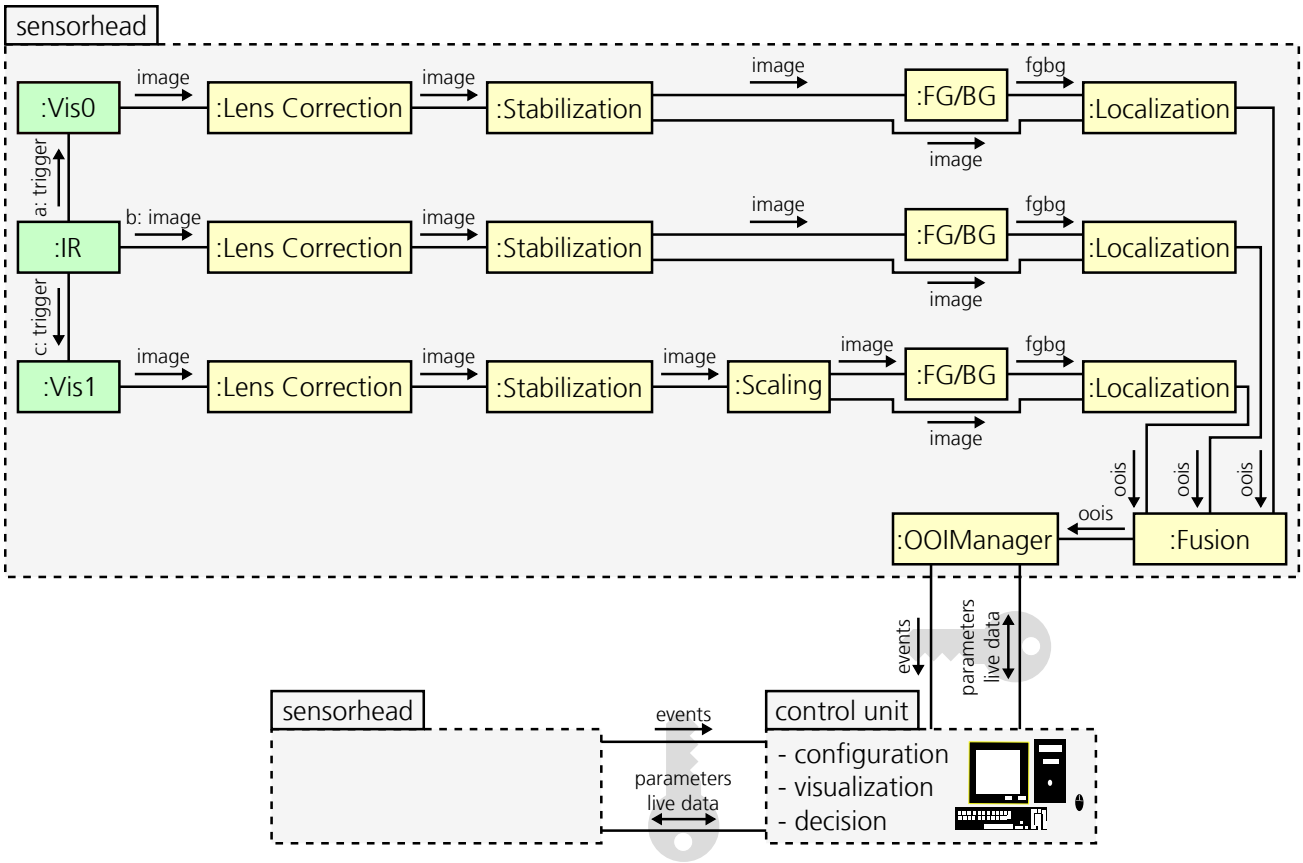

Figure 2. Software architecture overview of the demonstration set-up.

3. HARDWARE IMPLEMENTATION

The demonstration sensorhead housing, as shown in Figure 1, was designed with outdoor usage in mind, being waterproof and made of stainless steel. Two cameras with 8 mm C-mount lenses and, in case of the IR camera, a 12 mm wide angle lens, all facing in the same direction, are separated by a total base length of 45 cm. Such a wide base length was chosen to improve depth accuracy of a classic stereo vision approach, which was still considered viable at this point. In practice, smaller base length and thus smaller overall dimensions of the sensorhead would be beneficial due to decreased influence of wind on camera rotation and translation. Because of good transmission levels in the infrared range, the camera holes are covered by Germanium glass. If no IR vision was required, ordinary glass with anti-reflex coating could be used as well.

Computing components were chosen with self-sufficiency in mind. Being typically the biggest power consumer, choosing a low-power CPU was key. An Intel Pentium N3710 CPU with integrated GPU was combined with two 2 GiB DDR 3 RAM modules. A solar charge controller supplies the system with power from either a 200 Ah car battery or a 1 m^2, 150 W solar panel. Multiple sensorheads are supposed to communicate with a central control unit either locally via the built-in WLAN module, or remotely via LTE. Restrictions on the mobile data plan further motivate efficient data reduction. At a total power consumption of 12 W under full load, the system can operate self-sufficiently for at least seven days.

4. SOFTWARE IMPLEMENTATION

The main goal of this work was to design an efficient, portable and reusable toolkit for possibly multi-spectral surveillance use cases. For this purpose, all software components were developed in the Python 3 programming language using well-established open-source libraries.[9,10] An overview of the software architecture is given in

Figure 3. Calibration pattern with VSoC camera (left), IR thermal camera (center) and color camera (right). Calibration marks can be found and matched to a common coordinate system for each camera independently.

Figure 2 in the form of a simplified communication diagram. All image processing is performed within the sensorhead. The IR camera operates at a fixed frame rate and issues a hardware trigger to both the HDR (Vis0) and color camera (Vis1). Image processing for the individual views is performed in parallel by separate, independent processes. Communication both between the individual processes and with an external control unit is handled via Protocol Buffers[11] and ZeroMQ.[12] For each acquired image frame, the corresponding process executes a sequential chain of image processing steps, each encapsulated as a self-contained object with well-defined input and output. This way, individual steps can be easily substituted if need be. The parallel processes output OOIs (objects of interest), which generally consist of an arbitrary set of parameters, with matching ones being combined to multi-spectrum OOIs. In this case, they consist of a position in a common 3-D coordinate system. Using a set of rules, these OOIs are converted into use-case-dependent events, e. g. "object at GPS position", which are transmitted via a secure connection to a central control unit. This unit is used to manage possibly multiple sensorheads, visualize their outputs and assist a human operator in decision making based on the sensor data. Data reduction is very important at this point, especially if no direct WLAN connection to the sensorhead is available.

For demonstration purposes, the sensorhead shall observe a parking lot and notify about incoming and leaving people and cars. A similar set-up might be applied in a disaster response use case to distinguish people from debris e. g. during a flooding scenario. In the following, certain key sections of the processing chain will be discussed. This does not include foreground/background segmentation (FG/BG), as this step is already very well covered by "off-the-shelf" components of OpenCV.[13]

4.1 Camera registration

Stereo vision requires both the cameras' intrinsic parameters and their relative orientation to be known. Common camera calibration approaches require either multiple views of an object with known geometry,[14] which is difficult at the desired viewing distance, or at least some kind of camera motion (rotation and/or translation)[15] to perform self-calibration, which is not feasible. Furthermore, stereo calibration typically requires accurately finding homologous features in corresponding images of different cameras, which is not easily possible due to the lack of a general multi-spectral feature point detector and descriptor. However, two assumptions about a typical surveillance use case can be made to significantly simplify this problem:

1. Observed objects, such as pedestrians and cars, move predominantly directly on the ground and
2. the ground can be approximated as either a 3-D plane or a collection of non-overlapping 3-D planes.

This way, the problem of localizing objects in 3-D space was converted to the problem of localizing objects in essentially 2-D space or multiple 2-D spaces by sacrifizing the ability to localize flying objects and introducing distortions for very tall ones. Both of these have to be dealt with in the object localization step.

Projective transformations of coplanar points are defined by a 3×3 homography matrix H_p according to Equation (1).

$$\mathbf{x_p} = \begin{pmatrix} x_p \\ y_p \\ w_p \end{pmatrix} = H_p \mathbf{x_s} = \begin{bmatrix} h^p_{11} & h^p_{12} & h^p_{13} \\ h^p_{21} & h^p_{22} & h^p_{23} \\ h^p_{31} & h^p_{32} & 1 \end{bmatrix} \begin{pmatrix} x_s \\ y_s \\ w_s \end{pmatrix} \tag{1}$$

From this, a possibly overdetermined system of linear equations can be infered from at least four point correspondences to solve for H_p. This means that for each plane, at least four points have to be localized in pixel coordinates in all camera images and matched with the corresponding coordinates in the plane coordinate system. The resulting homography matrix H_p transforms points $\mathbf{x_s}$ in pixel coordinates to points $\mathbf{x_p}$ in plane coordinates. Figure 3 depicts images of such a simple calibration pattern. For the IR camera, small buckets of hot water were placed on the appropriate spots. This approach allows for estimating the position of objects on the plane from a single camera image and easily matching corresponding objects in multiple views by their coordinates in the common coordinate system. Measuring the world coordinates of such a small set of points on a ground plane is significantly simpler than coming up with a full calibration set-up.

Note that the cameras' intrinsic parameters do not have to be calibrated. Only lens distortion and, in case of color cameras, TCA (transversal chromatic aberration), have to be accounted for, because the aforementioned homography approach assumes cameras adhering to the pinhole camera model. Estimating distortion parameters offline and only applying a pre-calculated undistortion map at run-time is common practice.

4.2 Image stabilization

The sensorhead is typically mounted at raised elevation to maximize the observable area. At this point, it is naturally subject to wind, especially with a $1\,\mathrm{m}^2$ solar panel mounted on top. Thus, the image stream has to be stabilized to ensure robust background subtraction later on. In general, homographies can be used to model inter-frame motion, if either the observed scene is coplanar or if camera motion is restricted to rotation only. Dong and Liu have dealt with the resulting accumulation errors using an associate Kalman filter to model camera motion from consecutive frame homographies.[16] However, at almost 70 ms per 1 MPix image frame on a 3.6 GHz Core i5 CPU, even this real-time approach is too computationally expensive for our use case and low-power set-up.

Fortunately, the coplanarity assumption made in Section 4.1 can also help simplify the image stabilization problem. Only the ground, i. e. the image of a 3-D plane, has to be stabilized – slight distortions of objects above the ground plane can be tolerated, because they do not qualitatively affect the background subtraction model. Thus, instead of actually estimating camera motion, projective transformations between planes are computed. At least four feature points per ground plane are localized and matched in consecutive image frames to estimate a homography matrix H_s defining the projective transformation of the image of the plane in the current frame back to its image in the previous one. It is used to transform (parts of) the current frame such that the image of the modeled plane appears stationary, according to Equations (2), (3) and (4).

$$\mathbf{x_s} = \begin{pmatrix} x'_s \\ y'_s \\ w'_s \end{pmatrix} = H_s \mathbf{x} = \begin{bmatrix} h^s_{11} & h^s_{12} & h^s_{13} \\ h^s_{21} & h^s_{22} & h^s_{23} \\ h^s_{31} & h^s_{32} & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ w \end{pmatrix} \tag{2}$$

$$x_s = \frac{x'_s}{w'_s} = \frac{h^s_{11}x + h^s_{12}y + h^s_{13}}{h^s_{31}x + h^s_{32}y + 1} \tag{3}$$

$$y_s = \frac{y'_s}{w'_s} = \frac{h^s_{21}x_s + h^s_{22}y + h^s_{23}}{h^s_{31}x + h^s_{32}y + 1} \tag{4}$$

A robust feature point detector and descriptor (SURF[17] or ORB) is used to reliably identify ground points only. Even though the complexity of this approach increases linearly with the number of modeled ground planes, this is not an issue, because the number of ground planes is typically very limited and finding and matching feature points is significantly more computationally complex, anyway.

Alternatively, for scenes at a large distance ($> 50\,\mathrm{m}$), it can be argued that any possible wind-induced camera translation is insignificant compared to the effects of rotation. Thus, assuming no translation, a homography describes the mapping of an arbitrary, not necessarily coplanar set of points when rotating the camera around its center of projection. This way, the whole image would be stabilized instead of individual ground planes. Note that this approach might not always be possible for IR vision, because it requires the image data to have some kind of persistent texture.

4.3 3-D localization

As already mentioned in Section 4.1, by making reasonable assumptions about the nature of the surveillance application, the problem of localizing objects in 3-D space was converted to the problem of localizing objects on a plane or a set of planes, which boils down to a two-step process for each detected foreground blob:

1. Select a reference point for the foreground blob and
2. calculate world coordinates by applying the homography transformation H_p to the point.

The major advantage of this approach is that no two views of the same object have to be identified in order to estimate world coordinates, i. e. no potentially computationally expensive (multi-spectral) feature matching has to be performed.

Step 2 is straightforward. Using stabilized image coordinates (x_s, y_s) and each camera's initial calibration homography matrix H_p, world coordinates are calculated by applying H_p and normalizing, accordingly.

$$x_w = \frac{x_p}{w_p} = \frac{h^p_{11}x_s + h^p_{12}y_s + h^p_{13}}{h^p_{31}x_s + h^p_{32}y_s + 1} \tag{5}$$

$$y_w = \frac{y_p}{w_p} = \frac{h^p_{21}x_s + h^p_{22}y_s + h^p_{23}}{h^p_{31}x_s + h^p_{32}y_s + 1} \tag{6}$$

Equations (5) and (6) are applied to each reference point, but which reference point to choose? Because the homography will transform points significantly above ground as though they were projected onto the ground plane from the camera's point of view, a reference point must be as close as possible to that plane. Using the center point of the non-rotated bounding rectangle's bottom edge is both computationally cheap and produced reasonably accurate results.

$$\mathbf{x} = \begin{pmatrix} x \\ y \\ w \end{pmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{7}$$

$$y = \frac{y}{w} = \frac{f \cdot \cos\alpha \cdot Y - f \cdot \sin\alpha \cdot Z}{\sin\alpha \cdot Y + \cos\alpha \cdot Z} \tag{8}$$

$$\frac{\delta y}{\delta Y} = \frac{f \cdot Z}{(Y \cdot \sin\alpha + Z \cdot \cos\alpha)^2} \tag{9}$$

Consider a camera with focal length f mounted relative to the ground plane and tilted downwards at an angle α. Equation (7) describes the mapping of a 3-D point $\begin{pmatrix} X & Y & Z & 1 \end{pmatrix}^\top$ onto its image plane. Differentiating Equation (7) with respect to Y gives a measure for the change in the y-coordinate on the image sensor for an increase in distance Y at height Z (Equation (9)). For example, with a pixel pitch of $8.75\,\mu\mathrm{m}$ and focal length $f = 8\,mm$, with the sensorhead tilted downwards at angle $\alpha = 60°$ and mounted at height $Z = 10\,m$ and distance $Y = 30\,m$, $\frac{\delta y}{\delta Y} = 9.53\,\frac{pixel}{m}$, i. e. a change in distance of $1\,m$ would result in a change in the y-coordinate in the image sensor of about 9-10 pixels. This is better than the expected depth error in a classic stero vision set-up with $45\,cm$ baseline at a comparable distance, as was initially planned in the hardware design phase. Thus, at reasonable operating distances, localization accuracy mostly depends on the accuracy of plane calibration, background segmentation and, most importantly, assumption 2 in Section 4.1.

4.4 Further processing

To output multi-spectrum OOIs, the detected foreground objects of all three camera views have to be combined into one. This can easily be achieved by matching the closest OOIs according to their position in the common coordinate system. Each OOI can be assigned various parameters, such as size/area, temperature mean and range or a scaled-down version of the corresponding ROIs in color, HDR and IR image, respectively, masked by the output of the background segmentation step. This can be used by the central control unit for further feature detection and classification. If an activity was not detected in all three views, e. g. because the moving object did not exhibit a sufficiently high temperature gradient to bereliably detected in the IR image, corresponding ROIs can still be estimated using the common coordinate system. Furthermore, each OOI should be assigned an ID, so that a human operator at the central control unit can more easily follow the movements of a specific object. This would usually require robust feature description and matching to reliably distinguish e. g. different people from one another. However, this is not practical on the employed low-power system architecture. Instead, objects are tracked solely by their position in the common coordinate system, which is sufficient, as long as they do not get too close to each other. As object tracking was not key to the use case, there was no need for a more sophisticated approach.

5. CONCLUSION

An efficient, portable and reusable toolkit for possibly multi-spectral surveillance use cases was presented and employed in a self-sufficient processing platform. Image processing and data fusion of the multi-spectral cameras was performed directly on a low-power sensorhead, showing that under certain reasonable assumptions about the nature of the surveillance use case, powerful hardware is not strictly required. In particular, many practical surveillance use cases are to observe relatively flat surfaces, such as in urban areas or river surfaces, thus converting the originally stereo vision localization problem to solely transformations between planes. This approach allowed for making a trade-off between a possible loss in accuracy on the one hand, depending on the planarity of the observed scene, and lower required computational power and not having to perform extensive camera calibration on the other hand. Processing steps were encapsulated as self-contained, interchangeable objects, allowing for maximum flexibility to adapt the set-up for different use cases.

ACKNOWLEDGMENTS

The development was supported by the German Federal Ministry of Education and Research (BMBF) within the information and communication technology (IKT) research initiative for SME (KMU Innovativ), joint project "SmartFusionCam", grant number 01IS15030B. The authors of this paper are solely responsible for its content.

REFERENCES

[1] Jens Döge, Christoph Hoppe, Peter Reichel, and Nico Peter. A 1 megapixel hdr image sensor soc with highly parallel mixed-signal processing. 2015.

[2] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, volume 11, page 2. Citeseer, 2011.

[3] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[4] Suhad Lateef Al-khafaji, Jun Zhou, Ali Zia, and Alan Wee-Chung Liew. Spectral-spatial scale invariant feature transform for hyperspectral images. *IEEE Transactions on Image Processing*, 27(2):837–850, 2018.

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014.

[6] Cheng-An Chien, Cheng-Yen Chang, Jui-Sheng Lee, Jia-Hou Chang, and Jiun-In Guo. Low complexity 3d depth map generation for stereo applications. In *2011 IEEE International Conference on Consumer Electronics (ICCE)*, pages 185–186. IEEE, 2011.

[7] Peter Reichel. *Effizienter Einsatz von Bildsensoren mit integrierter Signalverarbeitung.* PhD thesis, Technische Universität Dresden, 2016.

[8] Werner Weber, Lex James, Arjan van Velzen, Bruno Vulcano, Micha Stalpers, Arend van de Stadt, Jens Döge, and Wilfried Rabaud. Peripheral devices for the right light. *LED Professional*, page 68 ff., March/April 2015.

[9] Travis Oliphant. NumPy: A guide to NumPy. USA: Trelgol Publishing, 2006–. [Online; accessed <today>].

[10] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[11] Kenton Varda. Protocol buffers: Google's data interchange format. Technical report, Google, 6 2008.

[12] Pieter Hintjens. *ZeroMQ - Messaging for Many Applications*. O'Reilly Media, 2013.

[13] Lili Guo, Dan Xu, and Zhenping Qiang. Background subtraction using local svd binary pattern. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 86–94, 2016.

[14] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, December 2000. MSR-TR-98-71, Updated March 25, 1999.

[15] Marc Pollefeys, Reinhard Koch, and Luc Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision*, 32(1):7–25, Aug 1999.

[16] Jing Dong and Haibo Liu. Video stabilization for strict real-time applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(4):716–724, 2016.

[17] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.