Towards on-the fly multi-modal sensor calibration

Jon Muhovič[1], Janez Perš[1]

Univerza v Ljubljani
[1]*Fakulteta za elektrotehniko, Tržaška cesta 25, 1000 Ljubljana*
jon.muhovic@fe.uni-lj.si

Abstract

The robustness of autonomous vehicles can be significantly improved by using multiple sensor modalities. In addition to standard color cameras and less frequently used thermal, multispectral and polarization cameras, LIDAR and RADAR are most often used sensors, and are largely complementary to image sensors. However, the spatial calibration of such a system can be extremely challenging due to the difficulties in obtaining corresponding features from different modalities, as well as the inevitable parallax arising from different sensor positions. In this paper, we present a comprehensive strategy for calibrating such a system using a multi-modal target, and illustrate how such a strategy could be upgraded to an fully automatic, target-less calibration that would rely on features of the scene itself to align at least small sensor offsets from the calibrated position. We find that a high-level understanding of the scene is ideal for this task, as this way we can identify characteristic points for spatial alignment of sensor data of different modalities.

1 Introduction

Multi-sensor systems are widely used for their robustness to adverse effects that might hinder one of the sensors, but not the others. Additionally, different modalities can produce richer information that allows easier or higher quality scene understanding. But a necessary requirement for such systems is the establishment of relative inter-sensor position and rotation, i.e. system calibration. While some sensors lend themselves to this problem quite well, several sensor modalities can prove problematic. Correctly aligning cameras with different fields of view that are not mounted in a stereo fashion or have visually very different modalities is an especially difficult task. Our proposed method tries to solve this by using a custom calibration target and computational approaches that cover a wide range of modalities and ensure that even very different types of sensors can be calibrated.

2 Related work

With the advent of research into self-driving vehicles, a large body of work has been done on calibrating multi-sensor systems. This is mostly LIDAR-camera calibration, but many works also address calibrating thermal and IR sensors with visual light cameras, mostly for enabling pedestrian detection in low-light circumstances.

While stereo camera systems are a tried and true method for calibration [1], many different approaches are still being developed for calibrating more heterogeneous systems. This stems from the fact that different sensors require different approaches, and not all calibrations methods are suitable for various combinations of sensors.

Many approaches for calibrating a LIDAR sensor with a visual light camera have been proposed: some of them [2, 3, 4, 5, 6] use planar targets to estimate the mutual position of the sensors, while others use information already present in the observed scene to estimate said position [7, 8] or even couple that approach with rich semantic information [9]. While most of these methods rely on geometric analysis of both sensors' properties, CNNs have also been used to extract the pertinent information from both sensors and estimate their relative positions [10].

Many methods and datasets are being developed for pedestrian detection in low-light scenes. This includes dataset such as LLVIP [11] that uses infrared images coupled with visible light images and KAIST [12] which contains RGB and thermal images. As discussed in Section 3.1, proper image alignment can only be achieved by using a beam splitter or by employing absolute depth measurement. Several methods have been proposed to align images of different modalities using external depth information [13, 14]. A newer approach to image alignment is also using neural networks to extract and align the features in the images automatically [15]. Authors Wang et al. [16] propose a GAN-based approach for refining weakly aligned image pairs.

A large body of work has recently been presented for semantic segmentation of urban scenes, primarily from an autonomous vehicle standpoint. For the purpose of autonomous driving, reliable detection of drivable surfaces and potential obstacles is crucial, even in disadvantageous circumstances. On the Cityscapes dataset [17], the state-of-the-art position is currently held by the transformer-based method SegFormer [18].

3 Proposed calibration strategy

Our approach is based on a target that can be easily detected in all modalities and is set up so that useful features can be extracted and used for calibration. The key is the use of LIDAR as the focal point of the calibration, due to its 360° field of view and absolute distance measurements. Our calibration procedure relies on detecting a physical target in all of the sensors and extracting features that will allow us to establish the relative positioning of the sensors.

3.1 The parallax problem

Calibrating cameras with absolute distance sensors such as LIDAR can prove difficult, but it is ultimately exactly solvable, i.e. when the precise rotation and translation parameters are established, the calibration will be correct for any observed point in space. Due to 2D projection that is inherent in cameras, alignment between multiple cameras does not work in the same way. In order to perfectly align two images, one of two conditions must be satisfied: either both cameras lie in the exact same point in space or the observed scene must lie infinitely far away. Otherwise, points at different distances from the cameras will map slightly differently, which can be observed as the parallax effect. While stereo camera systems use this property to estimate absolute distance, it presents an obstacle in multi-modal camera systems. This can be solved by using external depth measurements to correctly calculate the mapping between images, and this is exactly the advantage of LIDAR.

3.2 Target detection

Localizing the calibration target in visual spectrum cameras is relatively straightforward, as we are employing a standard asymmetric circle grid commonly used for camera calibration. Since the size of the target and the camera's intrinsic parameters are known, we can find the location of each point of the grid in the camera image using a perspective-n-point algorithm. This gives us an approximation of the relative position of the camera to the target. While detecting the calibration grid is important for calculating the intrinsic parameters of cameras, we use it as a way of reliably detecting the entire area of the calibration target. As the size of the target is known, we can extract the locations of its corners with no extra effort. If using a pinhole camera model, just connecting the corners with straight lines gives us the contour of the target in the image plane.

In order to extract the target position from LIDAR data, we have to find features of our target we can detect reliably. Since our LIDAR sensor is arranged in 16 beams and rotates continuously, we can arrange the scan points by beam index as well as by azimuth. The features we wish to detect in our calibration target are its edges. If we observe points along a single LIDAR beam, object edges should be characterized by a large difference in distance between neighboring points. By filtering out LIDAR scan by this criterion we can extract potential edges in the scene which should also include the edges of our calibration target as shown in Figure 1.

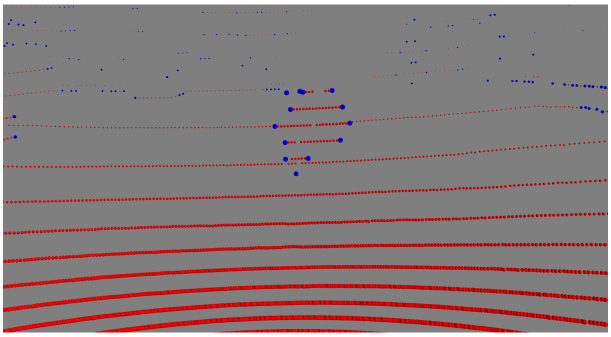

Figure 1: Lidar point cloud of the calibration target. Lidar points are depicted with red, while edge points are marked with blue. The target shape is clearly visible.

3.3 Position optimization

The problem of calibrating LIDAR sensors with cameras is the proper alignment of corresponding features. If the transformation between the sensors' coordinate systems is known, the alignment would be perfect. In that case, the features from both sensors would be aligned. This serves as a foundation on which the calibration can be optimized. In order to compare both modalities, edge features should be detected in both modalities, then the LIDAR points representing edges should be projected to the image plane.

A cost function that measures the alignment between edges in both LIDAR and image modalities can be formulated. In Section 3.2 we described the process of obtaining the image of target edges. Our cost function should measure how well the projected LIDAR points are aligned with said image edges. The cost function C can be formulated as:

$$C(I,P) = \sum_{i=0}^{N} I(P_i) \tag{1}$$

where N is the number of projected LIDAR points, I is the image of the target edges and P_i are the 2D coordinates of i-th LIDAR point. Since the Gaussian kernel is used for creating I, its intensities are highest on the target edges and the respective intensity gradient should allow convergence.

Using this cost function, standard gradient descent can be applied in order to maximize the alignment score. The parameters of the function are the rotation and translation components of the 4×4 transformation matrix used to transform the LIDAR coordinate system to the camera coordinate system. Knowing the camera intrinsic parameters, the LIDAR points can be projected onto the image plane and the cost function can be evaluated. Since the cost function cannot be analytically derived, numerical derivative estimation is used instead. A numerical derivative has to be calculated for each of the parameters at each optimization step. We use a central difference formula that approximates the derivative of our cost function as follows:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \tag{2}$$

where h is the step size.

The transformation matrix is formulated as a combination of the rotation matrix R and translation vector T and as such has 6 free parameters if we encode the rotation using Euler angles. The set of parameters for one LIDAR-camera pair can thus be represented as:

$$\Omega = [x, y, z, \vartheta, \varphi, \psi] \tag{3}$$

where the Euler angles that describe the rotation about each of the three axes are ϑ, φ, ψ, which represent roll, pitch and yaw, respectively.

Since a solution for a single viewpoint can be only locally correct, using several images with the target in different positions relative to the camera helps to generalize the solution. The approach is similar to using mini-batches in deep learning, as gradients for each image can be calculated at each step and the mean gradient can be used to update the solution.

3.4 Camera-camera calibration

Estimation of relative camera positions is a difficult task usually reserved for well set-up cases such as stereo systems where only offset on one axis is expected. Relative position estimation for arbitrary camera positions can be done by detecting a known object, calculating the essential matrix and decomposing it. But the procedure is not very numerically stable, nor is it guaranteed to produce the real calibration parameters even when using multiple target positions. Instead, we obtain the relative camera positions transitively after the respective LIDAR-camera calibrations have been completed. Given that each of LIDAR-camera calibrations is estimated using absolute distance measurements projected to the image, the margin for error is much smaller than if target position estimates were used. Let C_1 and C_2 be coordinate systems of cameras we wish to align and L be the LIDAR coordinate system. We will use the notation ${}^{A}H_{B}$ to signify the transformation from coordinate system A to coordinate system B. When calibrating C_1 and C_2 we calculate ${}^{L}H_{C_1}$ and ${}^{L}H_{C_2}$ respectively. We can formulate relative positions of cameras as follows:

$${}^{C_1}H_{C_2} = ({}^{L}H_{C_1})^{-1}\,{}^{L}H_{C_2}. \tag{4}$$

The mapping of pixels between images can then be realized by inverse projection as follows:

$$\begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \end{bmatrix} = K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \tag{5}$$

where K is the camera calibration matrix and u, v are the pixel positions. The inverse projection produces 3D coordinates $\tilde{X}$, $\tilde{Y}$ and $\tilde{Z}$ which are independent of the camera focal length. If this calculation is performed for all image pixels, this essentially creates a plane at distance $\tilde{Z} = 1$ in 3D space which, when projected back using K, perfectly reconstructs the original image. If we transform these points (call them $\tilde{I}$) to a different coordinate system, we can project them onto the image plane of another camera. However, this mapping will only be correct for objects at distance 1 from the original camera. Since points $\tilde{I}$ effectively lie on rays connecting some 3D point at unknown depth to the camera center, they can be multiplied by an arbitrary number, thus changing the simulated depth of the points with respect to the camera. This allows us to create maps between cameras at arbitrary depths. If we are only interested in observing objects at a specific distance from the cameras, this gives us perfect image alignment.

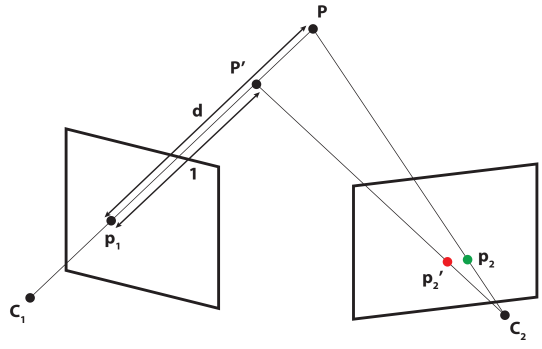

Figure 2: Depiction of image alignment process. If correct depth d of 3D point P is not known, correct pixel mapping can be only approximate, even if the calibration between cameras is known.

Given this formulation, it can be observed that each point can be moved away from the camera center independently. If some measurement or estimation of absolute depth is available for a pixel, alignment can be generalized to produce accurate results for any depth. This setup is shown in Figure 2, where the point p_1 on the image plane of the camera C_1 is *unprojected* into 3D space, generating the point p', which is then projected to the camera C_2 with some error based on the depth error. If, however, a more precise measurement of absolute distance d, such as a LIDAR sensor, is used, the position of the original point P can be better estimated and its projection p_2 will be positioned correctly.

It must be emphasized that this approach requires additional depth information if it is to provide accurate per-pixel alignment of two or more images. But it is geometrically the only viable option if our goal is to align more than two sensors, where accurate beam splitter approaches might become prohibitively difficult. While CNN-based feature alignment is possible, it is reasonable to employ as much accurate geometric approaches before using more data-driven methods, which can be highly susceptible to input data noise.

4 Experiments and results

We tested our approach on multiple modalities, but we present only RGB-polarization camera results. After each camera-LIDAR pair was calibrated, the images could be mapped into a joint viewpoint, as described

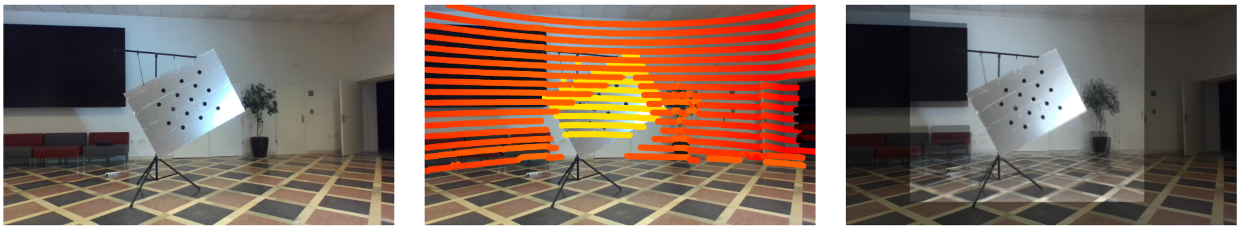

Figure 3: Mapping between ZED RGB camera, polarization camera and LIDAR, from left to right: raw RGB image, LIDAR data projected to RGB image, image from the polarization camera overlaid onto the RGB image.

in Section 3.4. We can then verify the correctness of our method by comparing the position of detected target points in the first image with the detected points in the second image and observe the reprojection error. Across the 15 images we selected for calibration, the average reprojection error for RGB-polarization setup is 7.6px.

4.1 Calibration in the wild

In complex multi-modal sensor systems, geometric decalibration due to slight sensor displacement or rotation while in the field is unfortunately a realistic scenario. In this case, the calibration target is not at hand, and cannot be used to correct the decalibration.

However, note that our approach relies on *detecting and segmenting a single object (target) in multiple modalities*. In the laboratory this could be a target of known dimensions with standard dot pattern, in the wild it could be any object that can be segmented in multiple modalities. We only present a visual illustration of this approach in Fig. 4. Note that the trees on the left side are perfectly segmented in LIDAR space, and slightly worse on the right. Semantic segmentation on RGB data is not perfect though, but this could be markedly improved using algorithms targeted to water environment, such as WaSR [19].

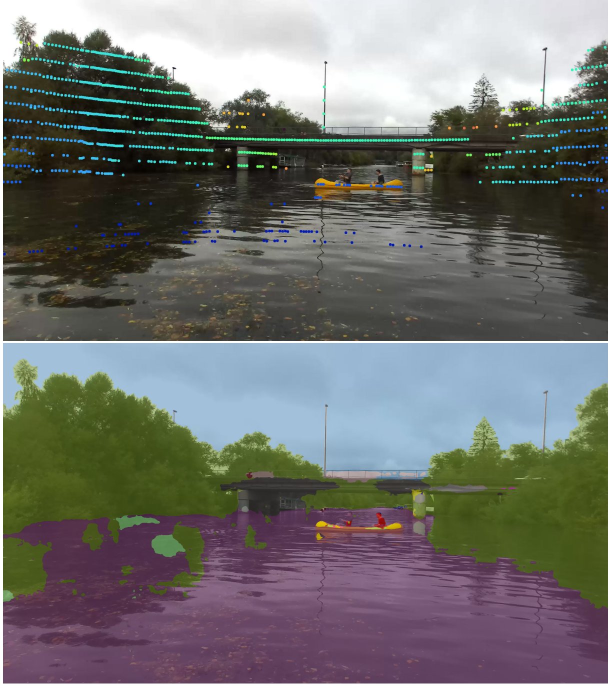

Figure 4: Multisensor data on the Ljubljanica river. Top: LIDAR data overlaid on RGB image. Bottom: semantic labels, obtained by SegFormer [18], overlaid on RGB image.

5 Conclusion

The presented calibration method solves multiple problems of multi-modal, multi-sensor calibration and works well in the laboratory conditions. We plan to extend this concept to the calibration in outdoor, target-less environment, where features from multiple deep architectures could be used to the same goal: finding reliable targets (objects) in images and LIDAR point clouds, using LIDAR distance measurements and proceeding with the proposed approach that works well in the lab.

Zahvala

To delo je bilo financirano s strani Javne agencije za raziskovalno dejavnost (ARRS), v okviru programa [P2-0095], in raziskovalnega projekta [J2-2506].

References

[1] H. Takahashi and F. Tomita, “Self-calibration of stereo cameras,” in *1988 Second International Conference on Computer Vision*. IEEE, 1988, pp. 123–128.

[2] E.-S. Kim and S.-Y. Park, “Extrinsic calibration of a camera-lidar multi sensor system using a planar chessboard,” in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2019, pp. 89–91.

[3] Z. Pusztai, I. Eichhardt, and L. Hajder, “Accurate calibration of multi-lidar-multi-camera systems,” *Sensors*, vol. 18, no. 7, p. 2139, 2018.

[4] K. Banerjee, D. Notz, J. Windelen, S. Gavarraju, and M. He, “Online camera lidar fusion and object detection on hybrid data for autonomous driving,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1632–1638.

[5] J.-K. Huang and J. W. Grizzle, “Improvements to target-based 3d lidar to camera calibration,” *IEEE Access*, vol. 8, pp. 134 101–134 110, 2020.

[6] C. Guindel, J. Beltrán, D. Martín, and F. García, “Automatic extrinsic calibration for lidar-stereo vehicle sensor setups,” in *2017 IEEE 20th International Conference on*

Intelligent Transportation Systems (ITSC). IEEE, 2017, pp. 1–6.

[7] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, “Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information,” in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[8] M. Á. Muñoz-Bañón, F. A. Candelas, and F. Torres, “Targetless camera-lidar calibration in unstructured environments,” *IEEE Access*, vol. 8, pp. 143 692–143 705, 2020.

[9] Y. Zhu, C. Li, and Y. Zhang, “Online camera-lidar calibration with sensor semantic information,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4970–4976.

[10] G. Iyer, R. K. Ram, J. K. Murthy, and K. M. Krishna, “Calibnet: Geometrically supervised extrinsic calibration using 3d spatial transformer networks,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1110–1117.

[11] X. Jia, C. Zhu, M. Li, W. Tang, and W. Zhou, “Llvip: A visible-infrared paired dataset for low-light vision,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3496–3504.

[12] S. Hwang, J. Park, N. Kim, Y. Choi, and I. So Kweon, “Multispectral pedestrian detection: Benchmark dataset and baseline,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1037–1045.

[13] J. Rangel, S. Soldan, and A. Kroll, “3d thermal imaging: Fusion of thermography and depth cameras,” in *International Conference on Quantitative InfraRed Thermography*, vol. 3, 2014.

[14] S. S. Shivakumar, N. Rodrigues, A. Zhou, I. D. Miller, V. Kumar, and C. J. Taylor, “Pst900: Rgb-thermal calibration, dataset and segmentation network,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9441–9447.

[15] C. Walters, O. Mendez, M. Johnson, and R. Bowden, “There and back again: Self-supervised multispectral correspondence estimation,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5147–5154.

[16] Y. Wang and D. Wijesekera, “Pixel invisibility: Detecting objects invisible in color images,” *arXiv preprint arXiv:2006.08383*, 2020.

[17] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[18] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, “Segformer: Simple and efficient design for semantic segmentation with transformers,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 077–12 090, 2021.

[19] B. Bovcon and M. Kristan, “A water-obstacle separation and refinement network for unmanned surface vehicles,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9470–9476.