Photogrammetric Multi-Camera Calibration Using An Industrial Programmable Robotic Arm

Laura Gonçalves Ribeiro

Tampere University
Tampere, Finland
laura.goncalvesribeiro@tuni.fi

Ahmed Durmush

Tampere University
Tampere, Finland
ahmed.durmush@tuni.fi

Olli Suominen

Tampere University
Tampere, Finland
olli.j.suominen@tuni.fi

Atanas Gotchev

Tampere University
Tampere, Finland
atanas.gotchev@tuni.fi

Abstract—Calibration of multiple cameras is a critical step in most vision enhancement systems. Target-based calibration approaches are known to provide accurate and stable results. However, they require manually performed capture procedures. This paper presents a generalization of a widely used single-camera target-based calibration algorithm to the case of n cameras. In order to obtain fully repeatable results, we propose the elimination of the manual capture step using a programmable robotic arm. Furthermore, we investigate the use of the position feedback provided by the robot. This is done specifically for the case of calibrating cameras without assumptions on their positions and overlapping of their fields of view. Results show that automatically captured images provide more accurate calibration results than the classical approach. Additionally, calibration of fully non-overlapping setups is made possible through our approach.

Index Terms—Multi-Camera, Calibration, Photogrammetric Calibration, Mobile Work Machines, Light-Field Capture

I. INTRODUCTION

Sensing is the starting point and limiting factor in any immersive visual reproduction of real-life content. Multi-viewpoint image sensing is required both for 3D scene reconstruction and image-based rendering. Current approaches to wide baseline multi-view image capture rely either on the use of a single moving camera or simultaneous capture using multiple cameras. In either case, and for virtually any application, cameras' internal parameters and positions need to be accurately estimated for adequate content interpretation and reproduction. However, while a single moving camera cannot capture dynamic scenes, the use of multiple cameras entails the calibration of an even higher number of camera parameters.

In this work, we look into methods for the calibration of multi-camera systems and analyze two particular sensor setups. First, we focus on the calibration of a light field capture system composed of 20 cameras in a linear equidistant configuration. In this setup, all cameras are approximately oriented in the same direction. The application that motivated this section is the setup of a teleconferencing environment using a light field display. These types of displays have relevant characteristics for teleconferencing due to the possibility of conveying directionality of the speaker's gaze and gestures, which provides important cues for communication.

Second, we focus on the calibration of camera systems where neither the cameras' fields of view intersect each other

Fig. 1. Concept of sensing around a work machine using a non-overlapping multi-camera setup.

nor any assumption can be made about their proximity in space. A relevant use case for this type of a setup is 3D scene sensing around work machines. In this application, cameras can be placed at any location and orientation to cover, for example, blind spots and enhance the operator's awareness of the machine's surroundings, as illustrated in Figure 1. The same principles apply to the calibration of other sensors that can be modeled through the pinhole camera model. This includes some types of time-of-flight and infrared cameras. Each sensor or sub-group of sensors may create a partial point cloud of a relevant section of the scene. These partial representations must be merged either for correct view rendering or extracting relevant measurements for operator assistance. Calibration parameters may also be used as additional constraints for generating maps through simultaneous localization and mapping (SLAM).

While the use of calibration targets is considered to be the most accurate and stable approach, the manual calibration image capture introduces variation to the results. Additionally, the higher the number of cameras in the rig, the more demanding and time consuming the calibration becomes.

Therefore we propose automatizing the image capture step of a standard photogrammetric calibration approach using an industrial robotic arm to manipulate a planar calibration object. With this method, we create an extensive dataset of calibration images with known relative target positions.

Our contribution to the state of the art can, thus, be summarized as follows:

- Implementation of an automatic calibration procedure

using a robotic arm.

- A data-set of calibration images with known relative target positions.
- An open-source implementation of a multi-camera calibration procedure.
- Development of a method for photogrammetric calibration of non-overlapping camera setups with arbitrary locations and orientations.

II. STATE OF THE ART OVERVIEW

Camera calibration is defined as the process of finding the transformation between the three-dimensional geometric location of a point and its corresponding point in the image plane of the camera [1]. It is performed through the establishment of a camera model and subsequent estimation of its parameters.

An ideal camera can be modeled by the pinhole camera model, where the focal length (f_x, f_y), principal point (c_x, c_y) and scaling factor (s) describe the mapping of 3-D world points into 2D image points through the principles of projective geometry, as described in equation 1. These parameters are referred to as *intrinsic parameters* or *intrinsics*. Real devices require, additionally, the estimation of the geometric distortions created by the introduction of a lens. Geometric distortion is frequently modeled using polynomials in the 2-D coordinates of the image plane which account for each of the radial and tangential components [2] [3]. The coefficients of these polynomials constitute the *distortion parameters*, which complement the set of intrinsic parameters.

Single camera calibration comprises the estimation of the camera's intrinsics and its position ($[R, t]$) (i.e. the *extrinsics*), in terms of a rotation R and a translation t, relative to a given world reference frame. The intrinsics and extrinsics describe the projection of world coordinates (X, Y, Z) into camera coordinates (x, y):

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (1)$$

Camera calibration is a widely researched topic. Existing approaches can be broadly divided into two groups: *photogrammetric calibration* and *self-calibration*.

Self-calibration relies on feature correspondences in arbitrary world scenes. This approach assumes that the scene is static and provides constraints on the camera parameters when observed from several viewpoints. A review of self-calibration algorithms can be found in [1] and [4]. This approach is particularly important in applications where cameras must be re-calibrated on the fly or are deployed to end-users and the use of calibration targets is not feasible.

Photogrammetric calibration relies on capturing a calibration object of precisely known geometry which acts as a space reference point. Approaches have been proposed using 3D calibration targets and 2D calibration targets undergoing known translation [5], [6]. Zhang proposed a more flexible technique using several (at least two) views of a planar target taken at different orientations [7].

Photogrammetric approaches are generally considered more accurate than self-calibration [8] as more constraints are taken into account and fewer variables are estimated at once. Additionally, detection of corners on calibration targets is known to perform more accurately than detection of image features such as scale invariant feature transform (SIFT) or speeded up robust features (SURF) [9]. Therefore, the use of a calibration target is the method of choice in applications with high precision requirements, such as in-factory camera calibration.

Zhangs' algorithm [7] has long been the gold standard for photogrammetric camera calibration and widely used toolboxes such as Bouguet's Toolbox [10], Matlab Camera Calibrator Application [11] and Open CV [12] implement it with slightly different distortion models. Its pipeline consists of the detection of feature points in the calibration images, estimation of the homography between a model point and its image and calculation of the closed-form solution of the camera parameters. Starting with the closed-form solution as an initial estimate, the mean squared error between the detected and projected points is minimized using the iterative Levenberg-Marquardt algorithm.

Multi-camera calibration consists of the estimation of each cameras' intrinsic parameters and the extrinsic geometric relation between their sensors. The extrinsic relations ($[R, t]_{c_n \to c_1}$) can be easily computed from the transformations between each camera and the same world reference target p_i ($[R, t]_{c_1 \to p_i}$, $[R, t]_{c_n \to p_i}$). Due to estimation errors, each calibration target image produces a slightly different value of the cameras' extrinsic parameters. Therefore, extensions of Zhang's algorithm to the stereo case calibrate, at first, each camera independently. This step is followed by the calculation of $[R, t]_{c_n \to c_1}$. As a different value is estimated for each image, the median is taken as an initial estimate. This estimate is refined by optimizing all calibration parameters over all target positions and both cameras. The extension of Zhang's algorithm to the two-camera case is implemented in such a manner in the abovementioned toolboxes. The overall optimization procedure is expected to improve independent calibration since the measurements from one camera might contribute to improving the estimation of other cameras' parameters [13].

There are few available implementations using a similar approach and taking into account more than two cameras. The AMCC toolbox [14] performs pairwise multi-camera calibration, i.e. results are optimized only between adjacent views. Herrera's toolbox [13] jointly calibrates a rig of several cameras and depth sensors. To the best of our knowledge, there is no readily available implementation of the extension of Zhang's algorithm to the calibration of more than two cameras of the same type with overall optimization.

When considering the joint calibration of many cameras, it is important to take into account the overlap of their fields of view. Figure 2 illustrates the case in which two cameras

have highly overlapping fields of view and an entire calibration object can be simultaneously detected by both cameras. Here, $[R,t]_{c_n \to c_1}$ can be directly calculated from $[R,t]_{c_1 \to p_i}$ and $[R,t]_{c_n \to p_i}$. This is often the case for arrangements with a small number of cameras. Particularly, if the cameras are parallel or face inwards. However, simultaneous detection of an entire target by an array of many cameras is frequently unfeasible. A higher number of cameras leads to a smaller shared field of view. Furthermore, the smaller the independent fields of view, the smaller the joint field of view of the system.

Fig. 2. Multi-camera setup with highly overlapping fields of view, where the calibration pattern can be fully seen by both cameras.

While most calibration patterns are composed of ambiguous features, some calibration patterns have been developed with unique features that make it possible to calibrate cameras, even if their fields of view have small or no overlaps. Examples include several modified versions of the checkerboard pattern, in which each corner in unambiguous, and more complex patterns like the one proposed by Li et al. [9]. This last target is obtained through reverse engineering of the SURF feature detection algorithm. SURF is an accelerated version of SIFT. SIFT extracts from images local features that are invariant to scale and rotation. The features are highly distinctive and are frequently used to solve image matching. The proposed target is constructed through the introduction of noise at different scales. Detected features are matched between the image and the known generated target image. The related toolbox can calibrate cameras with no overlap in their fields of view. However, it is limited to configurations where neighboring cameras can simultaneously observe different parts of the same calibration target, as illustrated in Figure 3.
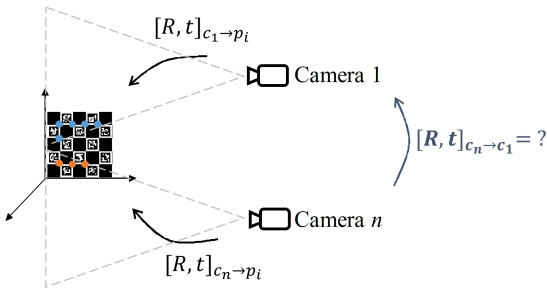

Fig. 3. Detection of unique target features by a camera setup with non-overlapping fields of view.

Another important requirement is that the feature points

should be distributed uniformly across the image. This is mainly needed for the adequate estimation of the distortion coefficients. [15]. In particular, it should be possible to detect feature points near the image boundaries, where radial distortions are generally higher.

Targets composed of unambiguous features are tackling also this problem since they make it easier to capture usable features close to the image borders. However, manual capturing introduces uncertainty in the feature distribution and makes the quality of results unpredictable. The authors in [15] address this problem by proposing the use of a computer screen spanning the camera's field of view and providing images of a virtual pattern. However, their solution cannot be efficiently applied to large camera arrangements.

Furthermore, we are not aware of any existing photogrammetric approach which would allow calibration of cameras in such a configuration that their fields of view do not overlap and it is not possible to simultaneously detect different regions of the same calibration target.

III. METHODS

A. Multi-camera Calibration

Calibration of n cameras imposes more constraints and cannot be accurately solved through independent and pairwise camera calibration. Therefore, in this section we propose a generalization of the Zhang's single camera calibration algorithm for the case of n cameras. More specifically, we run an overall optimization, which addresses all specified constraints for this case.

The calibration procedure goes as follows:

1) Independent single camera calibration using Zhang's algorithm. After this step, we obtain: an initial estimate of each camera's intrinsic matrix (K_n) and, for each calibration image where a target could be detected, the rigid transformation between camera and world reference frame ($[R,t]_{c_n \to p_i}$).

2) Estimation of each camera's position and orientation relative to a reference camera ($[R,t]_{c_n \to c_1}$). We use the left-most camera in the rig as a reference. As each pattern position provides a slightly different result, the median value is taken as an estimate of the extrinsic camera parameters.

3) Minimization of the reprojection error using the Levenberg-Marquardt algorithm. The reprojection error corresponds to the Euclidean distance between detected image feature points and reprojected world points, obtained through the principles of projective geometry. Figure 4 represents the inputs and output at each iteration step.

.

The minimization step runs over a subset of calibration images that can be simultaneously detected by all cameras. This set of images has, naturally, fewer features near the image edges. Thus, we hypothesize that the re-optimization of the intrinsic parameters, particularly of lens distortions, might not be desirable.
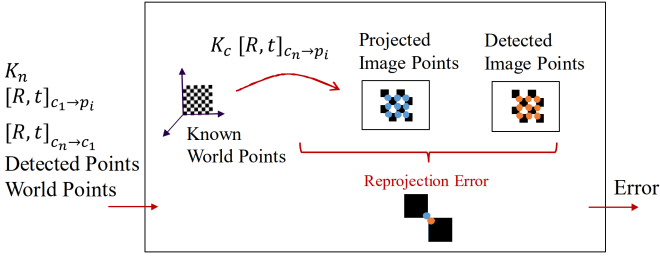
Fig. 4. Calculation of reprojection error. Inputs are: camera's intrinsic matrix (K_c), extrinsic relations ($[R,t]_{c_n \to c_1}$), world to reference camera transformations ($[R,t]_{c_n \to p_i}$), coordinates of world reference points and detected image feature points.

In the following sections, we will perform some experiments to evaluate which subset of camera parameters should be re-optimized for best results. In order to evaluate the calibration accuracy, we use the following metrics: 1) reprojection error in the calibration images; and 2) absolute vertical misalignment of target feature points in rectified test images. Test images are rectified pairwise in relation to one of the central cameras (in our arrangement, this is camera 10), and are not used in any of the calibration or optimization procedures.

B. Automatic Capture Using A Programmable Robotic Arm

In order to obtain a fully repeatable calibration pipeline, we implement an automatic capturing procedure using a robotic arm. The only manual step of this procedure is that the camera rig is initially positioned in such a configuration that the left-most camera sees the pattern. The pattern is rigidly attached to the robot end-effector. Additionally, practical considerations about the robot's working area need to be made. The robot motion that will produce the desired pattern-camera relations is calculated based on a rough nominal estimate of cameras' parameters and the transformation between the robot end-effector and calibration target. This hand to pattern transformation is obtained using the reprojection-based hand-eye calibration method proposed in [16].

During the procedure, two sub-sets of calibration images are captured. The first sub-set is composed of images taken at a distance of approximately 0,5 meters from the cameras. At such distance, the calibration target fills almost the entire camera field of view and thus facilitates the detection of more points close to the image borders. The second sub-set is composed of images taken at a distance of approximately 1 meter from the cameras at central positions, where all cameras can see the entire pattern.

C. Calibration Of Extreme Camera Configurations

In this section, we investigate whether the positions provided by the robotic arm can be used to calibrate two cameras while not assuming any spatial relation between them or their fields of view.

The calibration procedure goes as described in section A, except for a few modifications: The initial estimate of the

transformation between camera n and camera 1 ($[R,t]_{c_n \to c_1}$) is calculated from the relative positions of target position 1 (that can be detected by camera 1) and target position n (that can be detected by camera n), as illustrated in Figure 5. As shown in Figure 6, the relative target positions are calculated from the positions of the robot's end-effector relative to the robot center ($[R,t]_{r \to h}$) and applying hand to pattern transformation ($[R,t]_{h \to p}$).
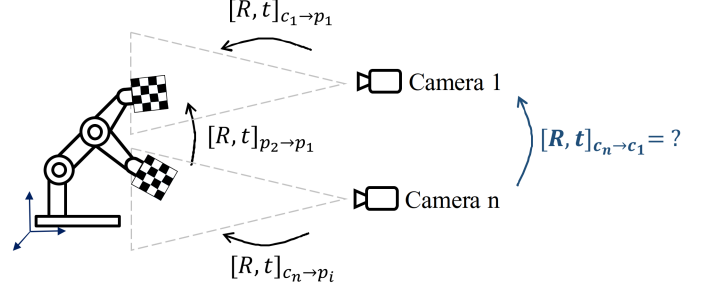

Fig. 5. Calibrating of two cameras in a setup where they cannot detect the same world reference target, however, relative target positions are known.
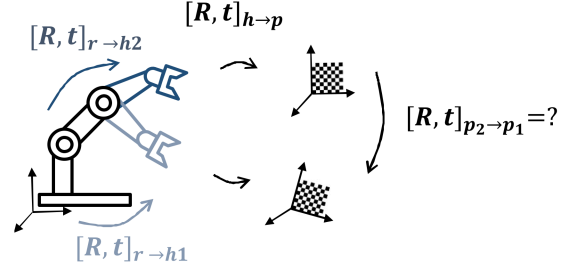

Fig. 6. Calculation of relative target positions based on the positions of robot's end-effector and hand-pattern calibration.

Calibration target positions are divided into two groups. Group 1 is visible by the reference camera. Thus $[R,t]_{c_1 \to p_{group1}}$ is calculated from image-based measurements. Group 2 is not visible by the reference camera. Thus, $[R,t]_{c_1 \to p_{group2}}$ is calculated from both image-based measurements and robot positions.

We implement an iterative optimization procedure. As a first step we optimize $[R,t]_{c_n \to c_1}$, while other variables are fixed. In a second stage we optimize $[R,t]_{c_1 \to p_{group2}}$ and $[R,t]_{c_n \to c_1}$ simultaneously. In the last stage we optimize $[R,t]_{c_1 \to p_{group1}}$ and $[R,t]_{c_n \to c_1}$. The choice of parameter optimization order is related with the confidence we have in each group of parameters. We expect estimates relying on a combination of image-based measurements, robot positions and hand-pattern calibration to be less reliable than sole image-based measurements. Inaccuracy in the robot positions and hand-eye calibration are expected to be significant and contribute to the uncertainty of the measurement.

IV. Experiments

We perform our experiments using a camera rig composed of 20 Basler acA1920-50gc GigE cameras with 1920x1200 pixel resolution rigidly attached to a metal structure in a linear equidistant configuration. These are connected through an Ethernet switch to the control computer. We use high quality 6 mm C-mount lenses. Their fields of view are, according to the manufacturer's specifications, approximately 84 degrees for the given sensor size.

We use the industrial robotic arm KUKA KR 16 L6-2. According to the manufacturer specifications, the robot has a repeatability of 0,05 mm. The manufacturer does not state the absolute accuracy of the robot. Potentially significant additional errors are expected if the attachment between the robot base and the floor is not sufficiently rigid.

The absolute accuracy of the robot was evaluated in the course of another work using a spherically mounted retro-reflector target and the Sokkia NET05 electronic distance measurement system [17]. Using the abovementioned methods, the accuracy of the robot was evaluated to be 0,3 mm on average with maximum errors of 0,8 mm.

The error in hand-pattern calibration is estimated to be 0.338 pixels.

In our work, we use most of the tools and functions provided by Matlab (R2017 version) and mostly follow their conventions. When performing single camera calibration, we consider only the first three terms of radial distortion, since we found that the tangential distortion was negligible for the high quality lenses we experimented with. The Matlab implementation of our calibration approach will be made available in the following link: https://immersafe-itn.eu/trainees/esr1-laura-ribeiro/.

The comparison between manual and automatic positioning of the calibration pattern is drawn by asking ten participants to perform calibration of the 20-camera rig. Participants were researchers from the laboratory and had some understanding of camera calibration. They were asked to take as many images as they deemed necessary to satisfy the following conditions:

- Each camera sees at least 3 fully visible pattern positions.
- Account for as much of each camera's image frames as possible.
- In 10 to 20 pattern positions the checkerboard is fully visible by all cameras.
- In each image, the checkerboard pattern must be at different orientations relative to the camera.
- The checkerboard should preferably fill at least 20% of the image frame.
- The checkerboard must be at an angle of less than 45 degrees relative to the camera plane.

These instructions are an adaptation of the Matlab documentation but for the multi-camera case.

V. Results

A. Linear Multi-camera Setup

We expect that adequately calibrated cameras should produce images that will be correctly rectified. Therefore, we
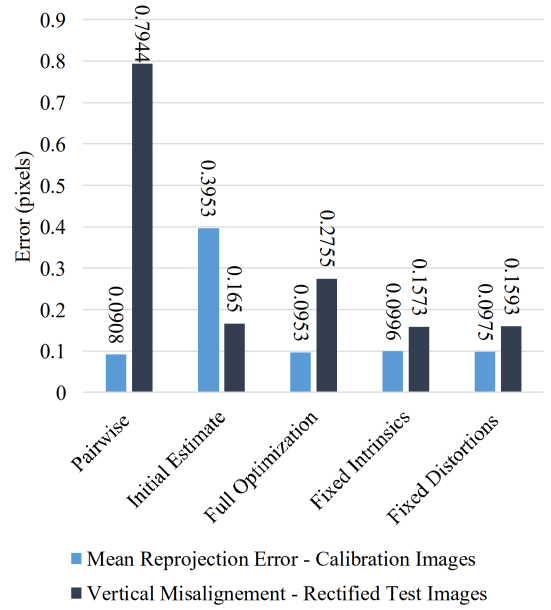

Fig. 7. Error in the calibration of the linear 20 camera rig.

quantify the calibration results not only by calculating the reprojection error by also by calculating the vertical misalignment after rectifying images, which were not used in the calibration procedure.

Figure 7 shows the error associated with the calibration of the linear 20 camera rig. We note that the reprojection error is lowest when all parameters are refined, i.e. full optimization. However, the vertical misalignment of rectified image pairs is lowest when the intrinsic parameters or the distortions are fixed during the optimization procedure. These results agree with our initial hypothesis. Additionally, we see that a low reprojection error on the optimization set does not necessarily mean that the estimated model parameters are correct or provide the best rectification results in other image sets. The refinement procedure might be over-learning the characteristics of the calibration images and provide poor results in real imaging conditions. Furthermore, our results are well within the sub-pixel range and perform considerably better than pairwise calibration.

Figure 8 presents the comparative results of the calibration performed by 10 participants and by the robotic arm. We observe that both quality measures are considerably lower when the pattern is manipulated by the robot. We also observe that there is significant variability among the results obtained by different participants.

B. Non-overlapping Stereo

In this experiment, we considered only the first and last cameras of the rig, and calibration images taken at such distance that the target cannot be simultaneously seen by the two cameras. This setting simulates the conditions of a non-overlapping setup. Figure 9 presents the results of calibrating
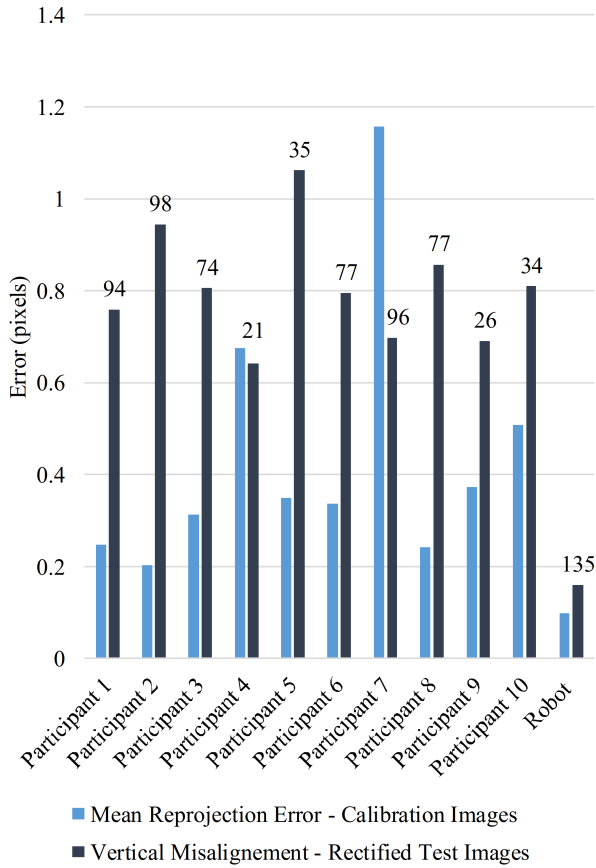
Fig. 8. Error in the calibration of the linear 20 camera rig. Numbers on top of the bars represent the number of images in each set.

Fig. 9. Error in the calibration of the non-overlapping stereo pair for each consecutive step of the optimization procedure.

the stereo pair with no overlapping features, at each optimization step. We observe that the initial parameter estimate is quite poor. This is likely due to the error associated with the robot positions. This initial estimate can be drastically improved by optimizing the extrinsic camera relations while fixing all other variables (optimization step 1). This improved estimate can be further improved by re-optimizing other groups of parameters, as described in the previous sections (optimization steps 2 and 3).

VI. Conclusions

The results of the calibration of the 20 camera system are satisfactory as we were able to achieve sub-pixel accuracy for both the calibration and test datasets. By using the robot arm we solved the repeatability issue of manual calibration. Additionally, we obtained more accurate results. Further work in this regard could include the use of targets with uniquely identifiable features. These targets would provide more feature points in less positions.

Results of the calibration of the non-overlapping stereo pair are also considered satisfactory. This type of setup can not be calibrated using standard photogrammetric calibration methods. The reprojection error is well in the sub-pixel range.
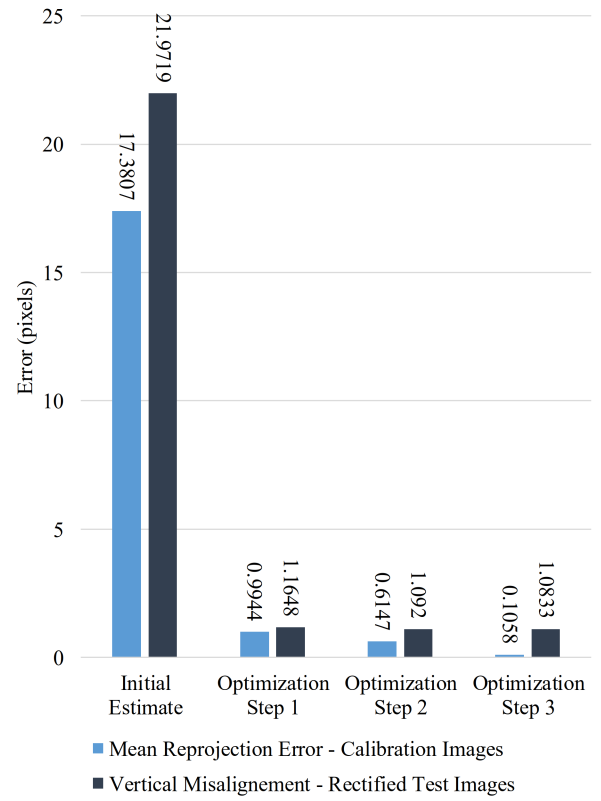
However, misalignment of the rectified test images is approximately 1 pixel. Further work could be done on improving the robot setup by compensating the positional errors.

It is important to note that these approaches provide an accurate estimation of the camera system's parameters in their current state. Vibration, movement or even changes in temperature might cause changes that deem the calibration parameters unusable. Strategies for the compensation of these changes or online calibration approaches might be needed to re-estimate these values in real imaging conditions.

References

[1] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003, doi: 10.1017/CBO9780511811685.

[2] L. Lucchese, "Geometric calibration of digital cameras through multiview rectification," *Image and vision computing*, vol. 23, no. 5, pp. 517–539, 2005, doi:10.1016/j.imavis.2005.01.001.

[3] C. B. Duane, "Close-range camera calibration," *Photogrametric Engineering*, vol. 37, no. 8, pp. 855–866, 1971, doi:10.1111/j.1477-9730.1976.tb00856.x.

[4] E. E. Hemayed, "A survey of camera self-calibration," in *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance*, 2003, pp. 351–357, doi: 10.1109/AVSS.2003.1217942.

[5] R. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987, doi:10.1109/JRA.1987.1087109.

[6] H. Malm and A. Heyden, "Extensions of plane-based calibration to the case of translational motion in a robot vision setting," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 322–333, 2006, doi:10.1109/TRO.2005.862477.

[7] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1330–1334, 2000, doi: 10.1109/34.888718.

[8] W. Qi, F. Li, and L. Zhenzhong, "Review on camera calibration," in *2010 Chinese Control and Decision Conference*. IEEE, 2010, pp. 3354–3358, doi:10.1109/CCDC.2010.5498574.

[9] B. Li, L. Heng, K. Koser, and M. Pollefeys, "A multiple-camera system calibration toolbox using a feature descriptor-based calibration pattern," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1301–1307, doi:10.1109/IROS.2013.6696517.

[10] J.-Y. Bouguet. (2004) Camera calibration toolbox for matlab. Retrieved 25.5.2019 from http://www.vision.caltech.edu/bouguetj/.

[11] Matlab. (2018) Camera calibrator app. Retrieved 25.5.2019 from https://se.mathworks.com/help/vision/ref/cameracalibrator-app.html.

[12] G. Bradski and A. Kaehler, *Learning OpenCV*. O'Reilly Media, Inc., 2008, doi:10.1109/MRA.2009.933612.

[13] D. Herrera, J. Kannala, and J. Heikkilä, "Joint depth and color camera calibration with distortion correction," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, 2012, pp. 2058–2064, doi: 10.1109/TPAMI.2012.125.

[14] M. Warren. (2015) Amcc toolbox. Retrieved 25.5.2019 from https://bitbucket.org/michaeldwarren/amcctoolbox/wiki/Home.

[15] L. Tan, Y. Wang, H. Yu, and J. Zhu, "Automatic camera calibration using active displays of a virtual pattern," *Sensors*, vol. 17, 2017, doi: 10.3390/s17040685.

[16] I. Ali, O. Suominen, A. Gotchev, and E. Ruiz Morales, "Methods for simultaneous robot-world hand-eye calibration: A comparative study," *Sensors*, 2019, doi:10.3390/s19122837.

[17] L. Hulttinen, "Kinematic calibration of kuka kr 16 l6-2," unpublished report, 2019.