New Methods of Eye-to-Eye Calibration and its Application to Cooperative Localization

Dem Fachbereich
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades
eines Doktor-Ingenieurs (Dr.-Ing.)
vorgelegte Dissertation

von

M. Sc. Zaijuan Li

geboren am 08. Nov 1988 in Henan

Referent:	Prof. Dr.-Ing. J. Adamy
Korreferent:	Prof. Dr.-Ing. U. Konigorski
Tag der Einreichung:	August 18, 2019
Tag der mündlichen Prüfung:	January 22, 2020

D17
Darmstadt 2019

Zaijuan, Li: New Methods of Eye-to-Eye Calibration and its Application to
Cooperative Localization
Darmstadt, Technischen Universität Darmstadt,
Jahr der Veröffentlichung der Dissertation auf TUprints: 2020
URN: urn:nbn:de:tuda-tuprints-114658
Tag der mündlichen Prüfung: 22.01.2020

Preface

The path to the destination is winding, but I am pleased that I finally made it. Along this thorny but significant journey, I learned and matured. I grew from someone who was always anxious, less confident in someone more patient, and never underestimates her strength.

I am grateful for my family. They support me in all the possible ways that a man could be supported. My husband Chao, the best gift I could ever imagine in my life, believes in me and never gives me up. I owe him hundreds and thousands 'Thank you. I love you.'. My parents are to me what a lighthouse is to sailors. They always guide me to the safe zone whenever I felt insecure.

I could not have achieved the current success without Professor Jürgen Adamy. His sincere support, consistent encouragement stimulate my potential and raise my spirits. I owe so much to Volker Willert, who helps me grow up both academically and mentally. From him,I learned to perceive a problem from a different perspective. I also learned to deal with pressure in the face of mental struggles. Most importantly, I have been given so much trust, freedom, and patience, based on which I thrive as time goes by.

I am thankful that I was always surrounded by coworkers, who create a friendly, relaxing, and conducive environment within which I could focus on my research. I received so much administrative support and help from Birgit Heid and Susanne Muntermann. Thanks to Valentina Ansel, who created all the beautiful, vivid graphics, the reader could understand the content of the thesis easier. I appreciate the efforts that Sylvia Gelman made to drag me out of lots of chaos that I caused in terms of all computer-related problems. I got so much academical inspiration and practical help from Raul, with whom I cooperate and like to cooperate a lot. The list goes on...

I also want to give my heartfelt gratitude to my dearest friends Xiaobo Mei, Xiaoying Hu, who never hesitate to help me with all sorts of difficulties.

I am who I am today because of all of you. I love you all and wish you all the best.

Contents

Abbreviations and Symbols

Abbreviations

ADAS	Advanced Driver Assistance Systems
BA	Bundle Adjustment
CRL	Cooperative Robot Localization
DLT	Direct Linear Transformation
DoG	the Difference of Gaussians
EKF	Extended Kalman Filter
EPnP	Efficient Perspective-n-Point
FOV	Fields Of View
KF	Kalman Filter
LiDAR	Light Detection And Ranging
LoG	the Laplacian of Gaussian
MOMA	MObile MArker based localization method
PF	Particle Filter
PnP	Perspective-n-Point
RANSAC	Random Sample Consensus
SAM	Structure And Motion
SE(3)	Special Euclidean Group
$\mathfrak{se}(3)$	the Lie algebra of SE(3)
SLAM	Simultaneous Localization And Mapping
S-MOMA	MOMA with Stereo camera
SO(3)	Special Orthogonal Group
$\mathfrak{so}(3)$	the Lie algebra of SO(3)
UKF	Unscented Kalman Filter
V-SLAM	Visual Simultaneous Localization And Mapping
VO	Visual Odometry

General Notations

$\mathbf{A}, \mathbf{B}$	the pose pair used for extrinsic camera calibration
$\mathbf{X}$	the unknown transformation between the calibration objects
$\mathbf{Y}$	the unknown transformation between the camera pair
$[,]$	binary operation
$\boldsymbol{\phi}$	Lie algebra vectors in $\mathbb{R}^3$
Φ	skew-symmetric matrix
$\boldsymbol{\xi}$	Lie algebra vectors in $\mathbb{R}^6$
$\wedge$	the operator transferring a vector into its corresponding matrix form
$\mathbf{K}$	the camera intrinsic matrix
$\mathbf{P}_i$	the i-th 3D feature point
$\mathbf{p}_i$	the 2D projection of the i-th 3D feature point $\mathbf{P}_i$
$C1, C2$	the camera frame *C1* and *C2*
$P1, P2$	the fiducial pattern frame *P1* and *P2*
$D1, D2$	the display frame *D1* and *D2*
$\boldsymbol{\varepsilon}_j$	the reprojection error of the j-th 3D-2D correspondence
$\boldsymbol{J}_{\boldsymbol{\xi}}^{\boldsymbol{\varepsilon}}$	the derivative of $\boldsymbol{\varepsilon}$ with respect to $\boldsymbol{\xi}$
${}^{B}\mathbf{G}_{A}$	the transformation $\mathbf{G}$ from frame A to frame B

Abstract

During the past few decades, an explosive development of multiple camera systems has occurred. For example, a multiple camera system can be used by an advanced driver-assistance system. For cooperative tasks among robots, a multi-camera rig can be used to increase the localization accuracy and robustness. In the logistics industry, a cargo drone mounted with a multi-camera system obtains a panorama view. In these or other high-demanding tasks that heavily depend on multi-camera systems, accurate extrinsic calibration of cameras is an absolute prerequisite for precise visual localization. In this dissertation, a weighted optimization method and a data selection strategy for extrinsic calibration are proposed that relieve the inherent imbalance between pose estimates existing in Liu's setup [39]. Besides, two new extrinsic calibration methods are proposed to improve the extrinsic calibration accuracy further. Other contributions of the thesis are two cooperative localization methods MOMA and S-MOMA, which can be applied to a robot group. These methods aim at overcoming the localization challenges in indoor environments where repetitive or lack of features are usually the case.

The weighted optimization method introduces a quality measure for all the entries of camera-to-marker pose estimates based on the projection size of the known planar calibration patterns on the image. The data selection strategy provides valuable suggestions on the selection of measurements leading to a better coverage in pose space used for the calibration procedure. By introducing a highly accurate tracking system, the first proposed calibration method disconnects the calibration objects, which are rigidly linked in Liu's setup. With the aid of the tracking system, the method improves calibration accuracy further. The second calibration method uses active calibration patterns realized with two electronic displays. By regulating the fiducial patterns displayed on the monitors, the approach can actively perceive the best possible measurements for the calibration estimation. The configuration of the dynamic virtual pattern aims at maximizing the underlying sensitivity of the objective function, which is based on the sum of reprojection errors, with regard to the relative pose between the camera and the fiducial pattern. State-of-the-art calibration methods, together with different configurations, are conducted and compared in simulation as well as in real experiments validating that both the optimization method and the two new calibration methods improve the calibration results in terms of accuracy and robustness.

In the second part of the dissertation, two novel, purely vision-based cooperative localization approaches MOMA and S-MOMA for a multi-robot system are introduced. MOMA realizes visual odometry via accurate MObile MArker-based positioning. The movement pattern of the robots mimics the movement of a caterpillar. The introduced fiducial marker board, which is mounted on one of the robots, serves as a mobile landmark, based on which the relative pose between the robots is recovered. The absolute positioning of each robot is deduced from the concatenation of the relative poses of previous phases. The second localization algorithm S-MOMA (MOMA with a stereo camera) extends the original MOMA approach. By fusing absolute pose estimates from static environment features with relative pose estimates from known mobile fiducial features, S-MOMA is formulated as an optimization problem combining two different objectives for these two different feature sources based on the same error measure, namely the reprojection error. A comparison between the proposed cooperative localization approaches MOMA, S-MOMA, as well as state-of-the-art localization algorithms for different configurations, is given validating the improvement in accuracy and robustness against various challenging testing environments.

Kurzfassung

In den letzten Jahren schreitet die Entwicklung neuartiger Multikamerasysteme rasant voran. Ein Multikamerasystem kann beispielsweise in einem intelligenten Fahrerassistenzsystem verwendet werden. Es kann auch für kooperative Aufgaben zwischen Robotern eingesetzt werden, um die Lokalisierungsgenauigkeit und Robustheit zu erhöhen. In der Logistikbranche kann eine Frachtdrohne über ein Mehrkamerasystem eine dreidimensionale Rundumsicht erlangen. Bei diesen oder anderen anspruchsvollen Aufgaben, welche erst durch Mehrkamerasysteme ermöglicht werden, ist eine genaue extrinsische Kalibrierung der Kameras notwendig, um eine präzise visuelle Lokalisierung zu erreichen. In dieser Dissertation werden eine gewichtete Optimierungsmethode und eine Datenselektionsstrategie vorgeschlagen, welche das inhärente Ungleichgewicht zwischen Posenschätzungen, das in Liu's Aufbau [39] vorhanden ist, weitestgehend aufheben. Außerdem werden zwei neue extrinsische Kalibriermethoden vorgeschlagen, um die Genauigkeit der extrinsischen Kalibrierung weiter zu verbessern. Weitere Beiträge der Arbeit sind zwei kooperative Lokalisierungsmethoden MOMA und S-MOMA, die auf einem mobilen Multi-Roboter-System angewendet werden können. Diese Methoden zielen darauf ab, die erschwerten Bedingungen bei einer visuellen Lokalisierung in Innenraumumgebungen zu überwinden, welche sich durch repetitive oder fehlende Merkmale ergeben.

Die vorgeschlagene Optimierungsmethode führt ein Qualitätsmaß für alle Kamera-zu-Marker-Posen-Schätzungen ein, das auf der Projektionsgröße bekannter planarer Kalibriermuster basiert. Die Datenauswahlstrategie extrahiert Bildmessungen mit besserer Abdeckung im dazugehörigen Posenraum als Eingangsdatensatz für die Kalibrierung. Durch die Einführung eines hochpräzisen Tracking-Systems können bei der ersten vorgeschlagenen Kalibriermethode die Kalibrierobjekte frei im Raum platziert werden und müssen nicht mehr wie in Liu's Anordnung fest miteinander verbunden sein. Dies führt zu einer Erhöhung der Kalibriergenauigkeit. Das zweite Kalibrierverfahren verwendet aktive Kalibriermuster, die aus zwei elektronischen Anzeigen bestehen. Durch eine Adaptation der auf den Monitoren angezeigten Referenzmuster während des Kalibriervorganges, kann der Ansatz aktiv bestmögliche Messungen für die Kalibrierung erzeugen. Die Konfiguration des dynamischen Musters zielt darauf ab, die Empfindlichkeit der nichtkonvexen Zielfunktion, die auf der Summe der Reprojektionsfehler basiert, in Bezug auf Posenänderungen zwischen Kamera und Referenzmuster zu maximieren. Desweiteren werden gängige Kalibriermethoden in Verbindung mit verschiedenen Konfigurationen sowohl auf simulierten, als auch realen Messdaten angewendet und verglichen, um zu bestätigen, dass sowohl die Optimierungsmethode als auch die neuen Kalibriermethoden die Kalibrierergebnisse in Bezug auf Genauigkeit und Robustheit verbessern.

Im zweiten Teil der Dissertation werden zwei neuartige, rein bildbasierte kooperative Lokalisierungsansätze MOMA und S-MOMA für ein Multi-Roboter-System vorgestellt. MOMA realisiert eine kooperative visuelle Odometrie über mobile visuelle Marker. Dazu werden spezielle Bewegungsmuster der Roboter benötigt, welche die Bewegung einer Raupe imitieren. Die visuellen Referenzmarkierungen werden auf einem der Roboter montiert und dienen als mobile Landmarke, anhand derer die relative Pose zwischen den Robotern hochgenau bestimmt werden kann. Die absolute Positionierung jedes Roboters ergibt sich aus der Verkettung dieser relativen Posen. Der zweite Lokalisierungsalgorithmus S-MOMA (MOMA mit Stereokamera) erweitert das Lokalisierungsprinzip von MOMA. Dazu werden absolute Posenschätzungen einer SLAM Methode aus statischen Umgebungsmerkmalen mit relativen Posenschätzungen aus bekannten mobilen Referenzmerkmalen fusioniert. Die Fusion wird in S-MOMA über ein kombiniertes Optimierungsproblem erreicht, das zwei verschiedene Ziele für diese beiden unterschiedlichen Merkmalsquellen, basierend auf demselben Fehlermaß vereint, nämlich dem Reprojektionsfehler. Die vorgeschlagenen kooperativen Lokalisierungsansätze werden in verschiedenen Konfigurationen mit aus der Literatur bekannten Lokalisierungsalgorithmen verglichen, um Verbesserungen in Bezug auf Genauigkeit und Robustheit in verschiedenen anspruchsvollen Testumgebungen zu bestätigen.

1 Introduction

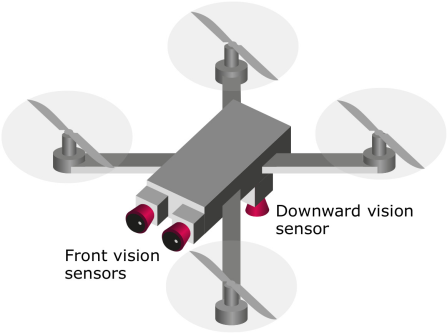


Figure 1.1: Sketch of a set of a front stereo camera and a bottom camera installed on a quadrocopter, which can be used for the automatic landing, localization, and mapping.

Due to the decreasing cost of manufacturing cameras, the past few decades have witnessed the explosive development of them. Cameras are everywhere, and they are 'infiltrating' into every possible aspect of our life. All cell phones are equipped with cameras. Cameras are installed along the highway for speed detection or within the building for surveillance and security reasons. Cameras with high resolution are used along production lines to assist quality tests or facilitate the assembly procedure. In some cases, multiple cameras are needed because one monocular camera fails to fulfill more and more sophisticated tasks. For example, in the robot community, a multi-camera platform provides more flexibility in sensor placement on a mobile robot for simultaneous localization and mapping (SLAM) [29]. A multi-camera system can also be used for 3D segmentation and cooperative localization among multiple mobile robots [42]. The majority of quadrocopters are equipped with at least two cameras (Figure 1.1) to perform automatic localization and landing tasks [4]. In the car industry, the multi-camera infrastructure can be used for computing loop-closure constraints [34], or be integrated to assist effective parking [63]. In the human-machine interaction area, a multi-camera system enables people to interact with the environment by building real-time 3D models [50] or could be applied to people-tracking for virtual reality television studios [48].

As we can see, it is becoming more and more ubiquitous in a variety of research fields to have multiple cameras with vastly different fields of view (FOV) mounted on a rigid object, such as a car, a mobile robot, a drone, etc. However, all these applications necessitate an accurate extrinsic calibration of the mounted cameras, which estimates the relative pose between these cameras. Moreover, the performance of these applications is highly correlated with the accuracy of the extrinsic calibration. We refer to the extrinsic calibration of cameras with disjoint FOV as eye-to-eye calibration to analogize the hand-eye calibration [28].

Precise and robust localization is recognized as one of the main fundamental requirements for mobile robot autonomy. Required tasks such as obstacle avoidance, path planning, and mapping [24], [65] could only be successfully conducted after accurate positions or poses of the agents are acquired. Different kinds of sensors could be equipped for localization: radar, lidar, laser rangefinder, infrared range sensor, camera, etc. For accuracy and robustness reasons, sensors are usually combined to compensate for the limitations of different sensors. Compared to other sensors, a camera has many advantages. It is light weighted, and the cost is lower. Besides, the image contains rich information about the environment. With the algorithmic development in computer vision and the increasing computing power which could afford the image processing in real time, the camera is becoming one of the most popular sensors for agents to perform combined perception and localization tasks. So in this thesis, new cooperative localization methods are explored, which are purely vision-based.

1.1 Motivation

1.1.1 Eye-to-eye Calibration

Eye-to-eye calibration estimates the relative pose between cameras with disjoint FOV. It is similar to the stereo camera calibration in the sense that both calibration procedures are relating the cameras with the same set of 3D features during the calibration process. Unlike the stereo camera calibration, where one calibration checkerboard is usually adequate to relate different camera frames, eye-to-eye calibration is dealing with cameras with non-overlapping FOV. Due to the non-overlapping FOV, the relating process, in this case, is very challenging since the introduced features captured by one camera will not appear in the FOV of the other camera. Borrowing the idea from the stereo camera calibration, one straightforward solution to get around this limitation is to introduce calibration targets, which have to be individually designed based on different configurations of the cameras.

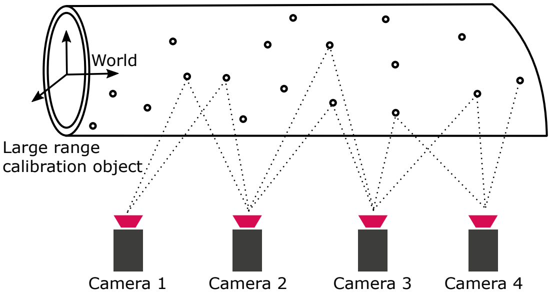


Figure 1.2: The principle of eye-to-eye calibration using one large calibration object.

The calibration object can be of various shapes or forms. One obvious way is to manufacture a large calibration object like the calibration item shown in Figure 1.2 so that each to-be-calibrated camera could detect some parts of it. The relative pose between the cameras could then be related. Unlike the stereo camera calibration, the size of the calibration object used for the extrinsic calibration is highly dependent on the configuration of the cameras. In some cases, the size of the calibration object can be huge such that all cameras could capture some features on it at the same time.

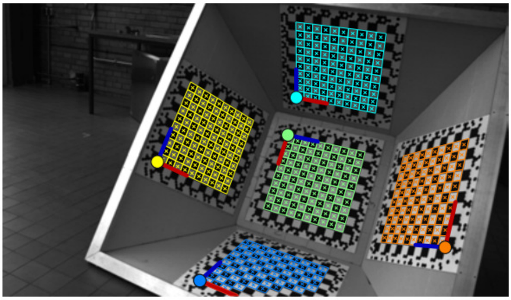


Figure 1.3: Strauss's calibration setup. The checkerboards are surrounded by the encoded binary code, which gives each board its unique indexing.

Another way is to construct multiple calibration objects, and the relative pose between them is kept unchanged during the calibration procedure. One example is to introduce multiple checkerboards surrounded by encoded binary codes so that each checkerboard has its unique board indexing [61]. The robot with

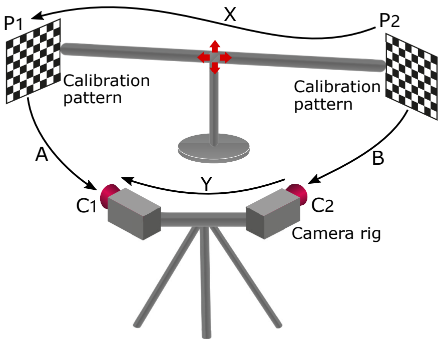


Figure 1.4: Liu's calibration setup. Two planar calibration patterns *P1*, *P2* are rigidly linked and fixed to a moveable frame. The frame is placed in several positions to the camera rig during the calibration process.

a mounted camera rig moves around the checkerboards, and each camera could build a map of the checkerboards (Figure 1.3). The unknown pose between the cameras is estimated after fusing their previously built maps. Another example is Liu's method [39]. The method solves the unknown transform between the camera pair by placing a compound target consisting of two rigidly linked planar calibration boards at several different poses relative to the camera pair (Figure 1.4). Compared to Liu's method, the method in [61] reduces the difficulty of establishing the relationship between the calibration targets and the vision sensors using an encoded binary pattern. In this case, the camera does not necessarily have to capture the whole planar calibration pattern to recover its pose. However, more pre-designed calibration objects are needed, which increases the calibration complexity and results in less portability.

In general, the calibration methods that introduce calibration objects with built-in fiducial features share the following properties. First, these methods are less prone to failure since features from calibration targets could be extracted with very high certainty and accuracy. Second, the well-detected features could be further applied to refine the calibration results, which is one major advantage of introducing calibration objects with prior built-in features. However, the calibration items generally need to be pre-designed and pre-constructed to fit the specific configuration of the to-be-calibrated cameras. Moreover, the limited viewing range resulting from finite-sized calibration targets could, in some cases, cause unstable results. In contrast, large calibration objects are cumbersome and even impractical sometimes.

Now we refocus on Liu's method. The method exhibits high flexibility and is very straightforward to implement. Meanwhile, the calibration cost is relatively low. Unfortunately, since both calibration boards have to appear in the FOV of the corresponding cameras for each pair of measured images, the pose change space is reduced. What is more, the resulting measurement quality is imbalanced[1]. These limitations will have a negative effect on the stability of the method [71]. From the standpoint of a comprehensive evaluation, Liu's method is highly suitable for many eye-to-eye calibration scenarios even though it does own the above inherent imperfection. The question is: if Liu's setup is applied, is there any approach to relieve this underlying instability? The answer lies in the optimization method and the measurement selection strategy proposed in this thesis.

As mentioned before, the limitations in Liu's setup result in the reduced pose change space and the imbalanced measurement quality, which lead to the instability of Liu's method. So another related question is: is there any possible variation on the setup configuration that could be applied such that these inherent limitations could be relieved?

Two possible solutions are proposed in this thesis.

The first solution is straightforward, which is to disconnect the calibration objects so that they could be independently placed to each camera. The relative pose between the calibration objects could be accurately recovered from a tracking system. In this case, the tracking system serves as an invisible link between the calibration objects. However, the disconnection comes at the cost of introducing a highly accurate tracking system, which is normally expensive. Hence, the method of using the tracking system is not preferable if the introduction of the tracking system is only intended to solve the eye-to-eye calibration problem.

Instead of using the printed planar fiducial markers or specifically manufactured calibration objects, the work in [2] [40] introduces a display screen to generate fiducial patterns. The fiducial patterns could be flexibly controlled to fit different situations. Inspired by these tempting advantages, the second solution replaces the fix-sized calibration boards in Liu's setup with electronic displays. After proper encoding of the fiducial patterns displayed on the screen, the camera could still recover its relative pose to the screen even if it a captures only a part of the screen. Therefore, the pose change space of this new setup is larger compared to Liu's setup. Moreover, it is possible to improve the measurement quality by actively manipulating the fiducial pattern. The question is: if the screen could actively display fiducial patterns, how to design these fiducial patterns so that the captured images can generate more accurate camera pose estimation?

[1] A detailed explanation for the term *imbalance of the measurement quality* is given in Subsection 3.2.2.

1.1.2 Cooperative Robot Localization

Apart from the applications which demand robots to cooperate for some tasks like in [41], the past few decades have witnessed gradually increased research on cooperative robot localization (CRL) [55]. Even though particular attention and costs have to be paid for the disadvantages coming along with multiple robot systems, such as the increased complexity of the coordination of robots, the involved management of communicating, and the complex resolution of additional measurements among them [54], the advantages of localizing a robot group are tempting and multifold. First, by viewing all robot members as one sole entity and exchanging the relative pose information between them, it is more likely to prevent every single member from getting lost [54]. Second, the localization is more accurate because a more massive amount of measurements are gathered, from which less-noised data would have more influence on the pose estimates. Another possible benefit is efficiency since each robot participates in the localization task [41].

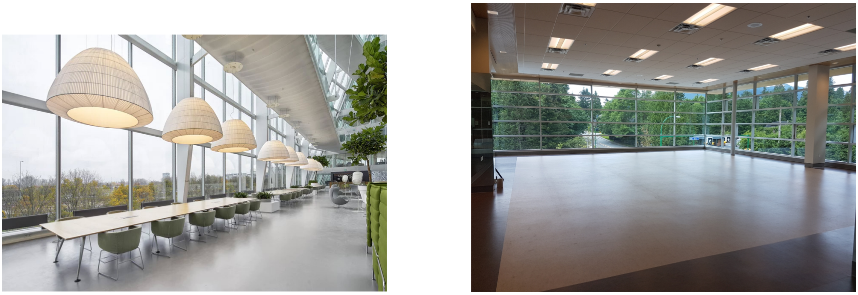


Figure 1.5: Examples of some challenging indoor environments in which visual localization methods are prone to failure. The environment in the left picture has repetitive features. In contrast, in the right picture, the room has a deficient number of features, which is inadequate for the robots to localize themselves successfully.

Compared to outdoor environments, localization methods are more prone to failure within indoor environments (Figure 1.5). Therefore, it is worth a moderate discussion on why localization within an indoor environment is more challenging compared to outdoors under most conditions. First, unlike in indoor GPS-denied situations, the positioning of outdoor agents equipped with GPS will always be guaranteed with bounded localization error regardless of which algorithm is applied [45]. Second, in indoor environments where unevenly distributed features, ambiguous features from artificial objects, and symmetric structures are usually the case, visual localization methods have to be robust enough to tackle all these potential problems.

On the one hand, simultaneous localization and mapping (SLAM) [18] methods are usually favored in indoor environments. Such methods estimate the egomotion of the agent while at the same time construct a 3D map of the environment and show significant advantages in reducing the positioning drift, especially when the agent is moving in a closed space. On the other hand, a visual odometry (VO) [49] based framework is preferable for the outdoor environment since building a global map is expensive, especially for large-scale areas. However, both SLAM and VO methods require rich features in the environment. Besides, they are vulnerable to erroneous feature matching, which would bring irrecoverable consequences on localization robustness and accuracy.

After combining the advantages of multi-robot cooperative localization methods and the challenges existing for indoor environment localization, the following core and reasonable questions are posed, which also highlight the contribution of the proposed localization methods in this thesis. How to guarantee the overall robustness when the environment does not possess a decent number of features to enable the robot positioning? How to improve the localization accuracy when there are unfavorably distributed features or ambiguous features in the environment? How to refine each robot's pose once it detects other robots or is detected by other robots in the circumstances mentioned above?

1.2 Contribution and Dissertation Structure

1.2.1 Contributions of the Dissertation

This thesis contributes to two research fields, namely, extrinsic calibration of cameras with non-overlapping FOV and cooperative localization of multi-agents. The contributions are summarized as follows.

- Firstly, an extended optimization method and a proper measurement selection strategy are proposed that can be integrated into specific calibration setups to enhance the accuracy as well as the stability of eye-to-eye estimates.

- Secondly, two new eye-to-eye calibration methods for several kinds of camera setups are presented.

- Lastly, the thesis introduces two new purely vision-based cooperative localization methods MOMA and S-MOMA, which can be applied to various challenging environments and show high accuracy and robustness.

MOMA is shortened for MObile MArker based localization method, and S-MOMA is developed based on MOMA but with an extra set of one stereo camera integrated.

1.2.2 Dissertation Outline

The dissertation is structured as follows. Since eye-to-eye calibration and cooperative localization are the two main contribution areas in this thesis, the following chapter begins with a brief sketch of different eye-to-eye calibration methods. Then a short but necessary introduction of the methods for solving the equation $\mathbf{AX} = \mathbf{YB}$[2] is given, based on which all the contributions concerning eye-to-eye calibration in this thesis are built. The review of the various cooperative localization methods is presented at the end of this chapter.

Chapter 3 introduces the weighted optimization method and the measurement selection strategy that can be integrated into specific calibration setups. Two applicable setups are introduced: Liu's setup as well as the one proposed in the thesis applying a highly accurate tracking system. To avoid the repetitions that exist in the deduction of these two setups, the optimization method and data selection strategy are explained using Liu's method. The improvements brought by the optimization method are validated both in simulation and in the real experiment.

The subsequent chapter 4 presents another new eye-to-eye calibration setup, which introduces two electronic monitors to display dynamic fiducial features. The mechanism of the dynamic fiducial feature generation is derived, followed by the optimization method of this new setup. Then the performance of the proposed method is compared to state-of-the-art methods under different configurations in simulation as well as in a real experiment.

In Chapter 5, the cooperative multi-robot localization method MOMA is first presented, based on which the localization method S-MOMA is explained and derived. The implementation of different experiment settings are demonstrated, and the results obtained from different methods and configurations are compared and illustrated.

In the last chapter, conclusions from the previous results are drawn, and an outlook for possible extensions is provided.

[2] The equation $\mathbf{AX} = \mathbf{YB}$ and its notation is explained in Section 2.2.

2 Literature Review

The first section of this chapter reviews the previous work of eye-to-eye calibration, which is the main content of Chapter 3 and Chapter 4. Then a short introduction of the methods for solving the equation $\mathbf{AX} = \mathbf{YB}$ is presented, since both the optimization method and the new calibration methods proposed in this thesis depend on the initial value of the relative poses $\mathbf{X}$ and $\mathbf{Y}$ from solving $\mathbf{AX} = \mathbf{YB}$. In the end, the cooperative localization methods from the past decades are concisely summarized, which provides preparation for Chapter 5.

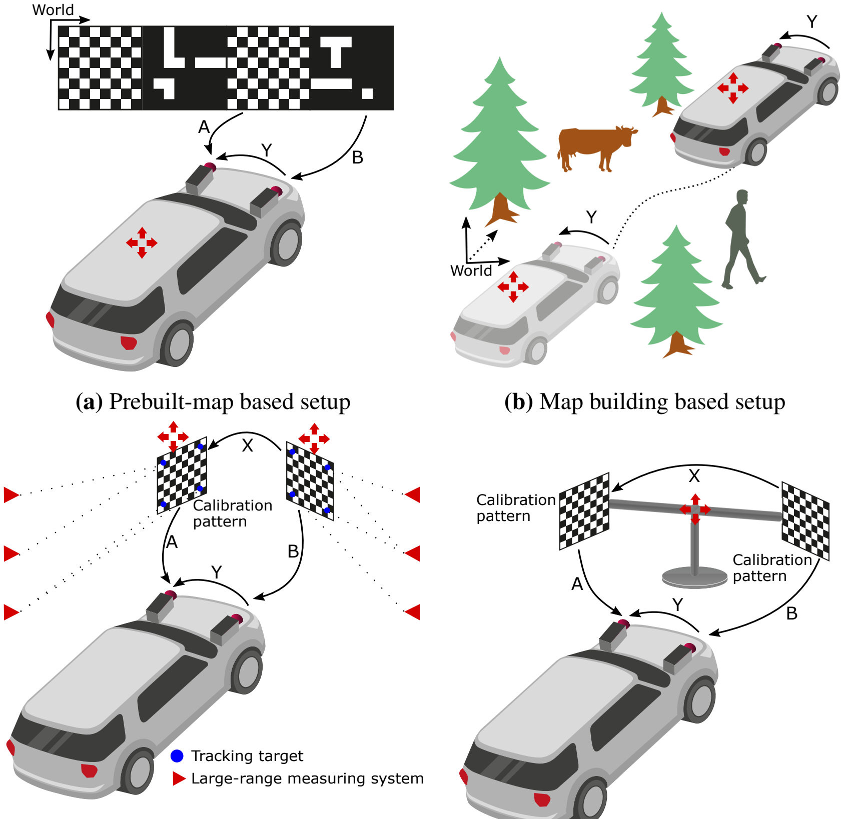
2.1 Previous Work in Eye-to-eye Calibration

Except for the eye-to-eye calibration methods explained in the last chapter, which apply larger-scale calibration objects, there are different methods for solving eye-to-eye calibration. In order to give a big picture of the existing calibration methods, it is necessary to categorize and compare these methods. There exist different category criteria. The most updated and detailed review could be found in [71], where the calibration methods are divided into six categories based on (1) Large-range measuring devices; (2) Large-scale calibration targets; (3) Optical mirrors; (4) Motion models; (5) Laser projection; (6) Visual measuring instruments.

In this thesis, the calibration methods/setups are classified into four categories I-IV shown in Figure 2.1 depending on (1) whether 3D features are mobile or not and (2) whether the absolute coordinates of the 3D feature points (either from a calibration pattern or the natural environment) are known or not during the calibration procedure.

Category I For the calibration of mobile multi-camera rigs using known absolute coordinates of 3D feature points, a 3D map of the environment is needed in order to localize the cameras within the map. Hence, an accurate 3D map of the environment [5], [26] must be available, or a reference object has to be placed in FOV of all cameras so that each camera could detect some parts of it [22], [13].

The methods in this category differentiate from each other in the way of how the map is constructed before the calibration procedure. The method in [5] ap-



(a) Prebuilt-map based setup

(b) Map building based setup

(c) Large-range measuring device based setup

(d) Liu's setup

Figure 2.1: Illustration of the calibration setup categories based on whether the features are mobile or not and whether the absolute position of the feature points is known or not during the calibration procedure. There are various calibration setups in each category. In order to emphasize the characteristics of the calibration setups in different categories, all the demonstrated examples are slightly reconfigured to calibrate a camera pair with non-overlapping FOV mounted on a car. The red arrows appearing in the figures indicate that the objects (vehicles or 3D features) with red arrows overlaid have to move or be placed in different locations during the calibration procedure. In the first calibration category, the car has to move within the previously well-built map to recover its localization (Figure 2.1a). In the second category, the car has to move in the natural environment with rich features in order that each camera could build an online map (Figure 2.1b). A large-range measuring system is introduced in the third category to detect the position of the fiducial features, which are used to relate different cameras (Figure 2.1c). Liu's setup is taken as an example for the last category. Two rigidly linked planar calibration patterns are introduced to build the relationship of the cameras (Figure 2.1d).

plied the RGB-D sensor to construct a global 3D model of the calibration environment. Heng et al. [26] used multiple cameras to perform a computationally expensive SLAM method to build a prior map of a calibration area, which could then be used for the recovery of the to-be-calibrated camera pose. The work in [22] constructed a global calibration object with circular calibration patterns pasted on, which could be detected by the camera. The positions of the circular targets' centers were obtained with the help of a hand-held scanner.

Category II When the camera rig platform is mobile while the absolute coordinates of 3D feature points are unknown, the calibration methods need either the recovery of each camera's relative pose between consecutive timestamps [19] or an online map building for each camera [10], [27], [33], [61]. The former method uses structure and motion (SAM) to formulate the eye-to-eye calibration problem, which is similar to the hand-eye calibration problem [28]. In contrast, the latter optimizes the unknown relative pose by aligning maps built by each camera, in which the maps could either be generated from a natural scene or a special calibration environment filled with fiducial landmarks.

The work in [19] formulated the problem similar to that of hand-eye calibration by matching the relative motion of multiple cameras in order to compute the extrinsics. The method only focused on the trajectory alignment, and a globally consistent map is neglected. Meanwhile, the method would degenerate under certain configurations [19]. In [61], Strauss etc. used multiple checkerboards surrounded by binary patterns to solve the association problem. The coordinate frame of different cameras and the boards, which could be partially detected by the camera due to the binary encoding, become mutually referenced over time. By additionally matching environment feature points and fusing the reconstructed maps from each camera, the work presented in [10], [27] addressed the degeneracy in [19]. In [10], Carrera G. et al. used MonoSLAM on each camera to build a globally consistent map. SURF descriptors, 3D similarity transform combined with a RANSAC framework were then applied to find inlier feature correspondences. In the end, a global bundle adjustment (BA) was run to optimize the camera pose, 3D feature points, and extrinsic parameters. However, the 3D similarity transform step might fail if the majority of the environment features are far from the cameras. As a result, the estimated 3D feature positions would contain substantial noise, which resulted in fewer inliers [27]. In order to maximize the number of feature correspondences, Heng et al. [27] used not only the current image but also a set of the most recent images from other cameras. The method additionally included an external motion sensor to recover the accurate scale of the map. Meanwhile, fiducial landmarks were introduced to refine the camera-odometry transform further. Besides, the methods in [10] and [27] shared one property that both rely on loop closures to maximize the robustness. However, the identification of the loop closure might fail in some cases.

The illustration of the above two categories is shown in Figure 2.1a and Figure 2.1b. The 3D features are represented as fiducial markers in Figure 2.1a to indicate that these features generate the most accurate and robust map compared to the features detected from the natural environment. For these two categories, the vehicles with mounted camera rig have to either move to several positions within the well-built map (Figure 2.1a) or move along in the scene to generate online maps (Figure 2.1b) for all onboard cameras.

Category III In this category, the camera rig is stationary. In order that the absolute coordinates of the 3D feature points from calibration patterns are known, a large-range measuring system as the one in Figure 2.1c is necessary such that the absolute position of the targets could be accurately tracked [11], [38], [37]. Then the unknown extrinsic could be solved based on (1) the accurately known absolute pose of the targets with reference to the global coordinate frame (the tracking system) and (2) their relative pose to the cameras. In a word, the methods in this category depend on either direct or indirect position information of the calibration targets.

Liu et al. [38] used a laser range finder to project laser spots on the planar calibration object and measure their distances. The extrinsics between cameras could be recovered according to the co-linearity of the laser spots, which have been captured by all cameras.

Category IV The last category deals with the configuration where the camera rig is stationary, and the absolute coordinates of 3D feature points are unknown. In this case, the 3D features are generally from reference objects and their relative positions are known. The reference targets are placed in different positions, and their pose with respect to the corresponding cameras is computed. These poses are later on used for recovering the extrinsic parameters.

In [39], Liu et al. introduced a moveable calibration rig with two rigidly linked planar calibration patterns. By changing the pose of the calibration rig relative to the camera pair, a set of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ was collected and used for solving for the unknown extrinsic camera pose (Figure 2.1d). Liu's method shows high flexibility, but the task of collecting a proper data set so that high-quality results could be generated is demanding since the calibration results are greatly dependent on the data set. The influence of different data sets on the calibration results will be explained in detail in the next chapter. Another way is to intro-

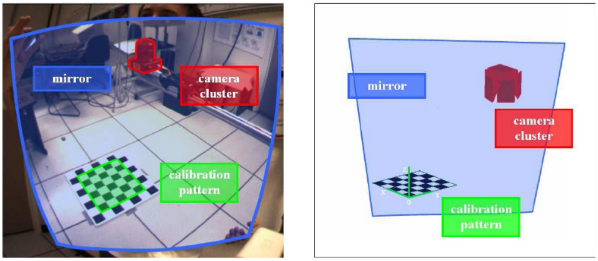


Figure 2.2: Kumar's calibration setup [30]. A mirror is placed in certain appropriate positions such that the calibration object, which is not originally in the FOV of the cameras, could be captured by its reflection. The right image gives the recovered camera pose using the calibration method.

duce a mirror to the calibration environment [30], [32]. The methods in [30] and [32] placed a mirror to several appropriate positions in order to image the calibration object that is not originally in the FOV of the camera (Figure 2.2). The extrinsic parameters of the camera were then calculated based on the mirrored views. Other similar methods belonging to this category could be found in [62] [40] [72], where sphere target, 1-D target, and particular target structure are constructed instead of planar ones.

There are no perfect setups. Every setup trades off between different optimization criteria and thus has its pros and cons. The methods [5], [26] in the first category (I) need a pre-built map, and they are advantageous only if many calibrations are needed within a short time span since the 'calibrated' environment must be kept unchanged afterward. Meanwhile, the multi-camera rig must be placed in the known map. Though the calibration approaches [10], [27] in the second category (II) are automatic and do not need a traditional calibration setup, they share the following problems. First, they have difficulty recovering the true scale of the environment or need an extra motion sensor. Second, an environment full of distinct features is implicitly required to guarantee the accuracy and robust outlier rejection strategies have to be carefully applied. Otherwise, the map-building process may fail. In addition, the accuracy is greatly dependent on the accuracy of the map, which inherits all the problems of map building methods like SLAM. The methods [33], [61] in the same category require moving cameras to capture the pattern boards at different times, which might be challenging for larger vehicles. The methods [11], [38], [37] applying a large-range measuring system in the third category (III) are generally more accurate, but the setup complexity and the costs are high. [30], [32] in the last category need to place the mirrors and grids to certain positions so that cameras could simultaneously detect direct or reflected calibration patterns. Though the techniques are easy and simple, the accuracy degrades as the distance between cameras becomes larger. Besides, the placement of the mirror is not straightforward to realize. Although the method [39] in the last category (IV) needs extra infrastructure and additional interaction is needed to collect measurement data, the calibration patterns could be detected reliably with sub-pixel accuracy, which provides true scale information and could be further included into the optimization process. Meanwhile, the setup complexity is low, and the costs are much less than buying and setting up a stationary large-range measuring system. Moreover, the camera rig does not have to be moved during calibration, which is a considerable advantage, especially for mobile vehicles. However, the limited pose change space of the calibration targets could result in low-quality calibration results [71].

	Stationary features	Mobile features
Known absolute 3D feature positions	Ⓘ Accuracy: high Robustness: high Portability: low Automation: low Price: medium Setup Complexity: high	(III) Accuracy: high Robustness: very high Portability: low Automation: high Price: very high Setup Complexity: very high
Unknown absolute 3D feature positions	(II) Accuracy: medium Robustness: low Portability: high Automation: medium Price: low Setup Complexity:low	(IV) Accuracy: medium Robustness: high Portability: medium Automation: medium Price: low Setup Complexity: medium

Table 2.1: Categorization and rating of different eye-to-eye calibration setups.

At the end of this section, a rating in terms of six practical assessment criteria that should be normally taken into consideration during the calibration procedure is included for each category, namely accuracy, robustness, portability, automation, price, and setup complexity. A summary of all the categories, including the rating, is given in table 2.1. Since there are various methods/setups in each category, the rating is a general, comprehensive evaluation of the category instead of a specific setup. Several conclusions could be drawn. First, when the accurate coordinates of the 3D features are known either from a highly accurate tracking system, a particularly constructed large calibration object, or a well-built map using high-quality sensors, the calibration accuracy and robustness are readily guaranteed at the expense of high costs, poor portability, and increased setup complexity. Second, when the coordinates of 3D features are not available and have to be recovered during the calibration procedure, the cost is generally much lower since no costly device is required. However, the accuracy and robustness are, in this case, compromised.

2.2 Review of the Methods for Solving $\mathbf{AX} = \mathbf{YB}$

In this thesis, the final estimation of the unknown pose transform $\mathbf{X}$ and $\mathbf{Y}$ is acquired from a nonlinear refinement, which is conducted after obtaining the initial estimation of $\mathbf{X}$ and $\mathbf{Y}$ from solving the equation $\mathbf{AX} = \mathbf{YB}$. Therefore, this section reviews the classical methods for solving $\mathbf{AX} = \mathbf{YB}$. All the transforms in the above equation are of the matrix form: $\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$, in which $\mathbf{R}$ stands for a 3×3 rotation matrix and $\mathbf{t}$ a 3×1 translation vector.

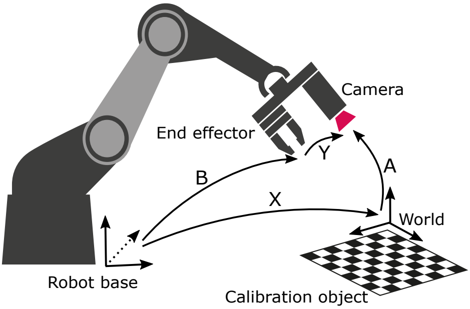


Figure 2.3: Hand-eye robot-world calibration.

$\mathbf{AX} = \mathbf{YB}$ is first proposed for solving the hand-eye robot-world calibration problem (Figure 2.3), where $\mathbf{X}$ represents the unknown transformation from the robot-base coordinate frame to the world coordinate frame, $\mathbf{Y}$ denotes the unknown transform between the hand frame and the camera frame, $\mathbf{A}$ is the transformation from the world system to the camera system, and $\mathbf{B}$ describes the transformation of the robot-base frame to the hand frame and is assumed to be known from the robot controller.

Same as the hand-eye robot-world calibration problem, the setups that are going to be presented in the following chapters also formulate the eye-to-eye calibration problem as $\mathbf{AX} = \mathbf{YB}$. For example, the transformation notations in Liu's

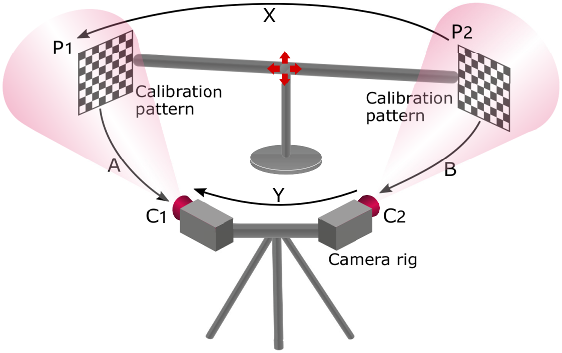


Figure 2.4: Eye-to-eye calibration.

setup are described as follows (Figure 2.4). **X** represents the unknown relative pose between two planar calibration patterns, **Y** represents the unknown transform between the camera pairs, **A** and **B** are the transformations between the camera and the corresponding planar calibration pattern, which are obtained after applying a marker detection algorithm. These transformations might be slightly different depending on different setups, but all the measurements are generated from cameras in eye-to-eye calibration setups.

The closed-loop constraint $\mathbf{AX} = \mathbf{YB}$ could be extended to (2.1), which can be decomposed into a rotational component (2.2) and a translational component (2.3) like follows:

$$\begin{bmatrix} \mathbf{R}_A & \mathbf{t}_A \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_X & \mathbf{t}_X \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_Y & \mathbf{t}_Y \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_B & \mathbf{t}_B \\ 0 & 1 \end{bmatrix} \tag{2.1}$$

$$\mathbf{R}_A \mathbf{R}_X = \mathbf{R}_Y \mathbf{R}_B \,, \tag{2.2}$$

$$\mathbf{R}_A \mathbf{t}_X + \mathbf{t}_A = \mathbf{R}_Y \mathbf{t}_B + \mathbf{t}_Y \,. \tag{2.3}$$

The methods for solving the above equations could be classified into the following three categories: separable solutions, simultaneous solutions, and iterative solutions. As the name suggests, separable solutions estimate **X** and **Y** by separately solving the rotational and translational components. The rotation part could be directly solved without involving the translation part. Then linear methods could be applied to solve $\mathbf{t}_X$ and $\mathbf{t}_Y$ once $\mathbf{R}_Y$ is known. In [73], the rotations were represented as quaternions and solved applying the linear solution, then the method of linear least squares is used to find the optimal translation part. Dornaika and Horaud in [17] proposed a closed-form solution for rotation estimation without the normalization process in [73], while the method of estimating translational components stayed the same. In [56], Shah applied the Kronecker product and singular value decomposition to find the solution.

The simultaneous solutions calculate **X** and **Y** by solving the rotational and translational components as a whole. Li et al. [36] used dual quaternions and the Kronecker product to simultaneously search for **X** and **Y** in order to limit the error propagation.

The iterative methods estimate **X** and **Y** iteratively. In [68], Wang et al. proposed a linear, approximative, iterative method to solve the rotation part using a variation of rotation matrices, and the translational component is solved in closed form.

2.3 Related Work in Cooperative Robot Localization Methods

This section reviews the related work in cooperative robot localization (CRL) methods. To simplify and clarify the recent work on cooperative localization methods, depending on whether the environment measurements have been employed to influence and bias the localization results, the cooperative localization methods are categorized into the following two groups: *environment-noninteractive* and *environment-interactive*.

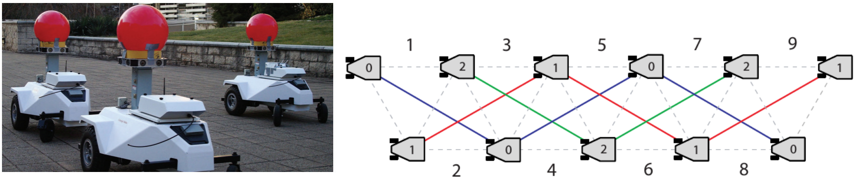


Figure 2.5: The leap-frog localization strategy from Tully et al [66]. The left figure shows three robots used for the experiment, and the right figure demonstrates the leap-frog path.

As the name suggests, the robots in the first category could only sense the bearing or positioning of other group members, and no measurements from the environment are included for localization estimation. An example belonging to this category was first introduced in [31] with the name cooperative robot localization (CRL). In this work, the robots were divided into two groups and alternated the roles of moving and staying static. The robot group, which remained motionless, acted as portable landmarks. This procedure repeated until both groups reached their destination.Tully et al. [66] designed a ‘leap-frog’ path for a team of three robots applying Extended Kalman Filter (EKF). In this case, two robots act as stationary measurement beacons, while the third moves in a path that provides bearing-only measurements. Similar to [31], the roles of each robot are switched, and the path is repeated (Figure 2.5). Luis et al. [45] fused the vision-based bearing measurements among pairs of robots with the motion of the vehicles by applying a recursive Bayes estimator. In [57], the authors presented a similar idea like in [45] while replaced the estimator with a non-linear counterpart and introduced additional fiducial landmarks to the environment in order to ensure the observability of the designed system. The drawbacks, however, reside in the limited exploring range and the inconvenience caused by the invasion of the fiducial marker. A bunch of similar approaches could be found in [15], [12], [23].

The environment-noninteractive localization methods are very robust and do not involve complex control management. However, they share the following limitations. Either at least one robot should keep stationary during the whole localization process, or fiducial landmarks must be introduced and appear in the FOV of the equipped sensors. The former slows down the overall localization procedure while the latter needs to transplant unfriendly human-made markers into the environment beforehand. The robot team is ‘blind’ to the environment, which means either the localization results serve as the input into higher-layer tasks, or the robot team is controlled by the operator. Since no map is built in this category, there exists unavoidable drift. Moreover, the longer the trajectory is, the larger the resulting drift will be.

As for the environment-interactive category, it is acknowledged that including environment measurements can improve the efficiency and accuracy. In [20], Fox et al. proposed a probabilistic Markov localization approach for a multi-robot system. The belief of each robot’s pose uncertainty would be biased when it was detected by other robots, or it detected other robots depending on the quality of the sensor for the detection. Though at the cost of the communicational overheads among robots, the approach showed drastic improvements in localization speed and accuracy. The main drawback is a known-environment map must be previously provided, which limits its applications in the real environment. In [54], a centralized estimation approach applying Kalman filter (KF) was presented using two cycles: the propagation cycle and the update cycle. The centralized estimator could be decomposed into a decentralized form, which allowed the measurements collected from a variety of sensors to be fused with minimal communication and processing requirements. Similar approaches were explored in [16], [8], [44], [9].

Some shared limitations of the environment-interactive category are as follows. The methods are less stable since the measurements from the environment are far less reliable and predictable compared to the relative measurements between the robot members. Both the communication mechanism and the measurement fusion are much more complex, which introduces increased overheads. Meanwhile, the management of the uncertainty distribution of each robot pose becomes complicated. However, all these methods do not need fiducial features, so there is no invasion of the fiducial landmarks to the environment. Besides, the map built during the localization process relieves the drift, especially when the robots are moving within an enclosed space, and the map also allows the robots to carry out more advanced tasks such as obstacle avoidance, path planning, etc.

The fusion algorithm in the second category could behave in either a centralized or decentralized manner during information exchanges. In the early developed methods, the fusion strategy was mostly filter based by applying filters such as extended Kalman filter (EKF), unscented Kalman filter (UKF), particle filter (PF). With the drastic improvement in computation power nowadays, the localization algorithms have shown the tendency of developing from the filter-based framework to the non-linear optimization-based fashion [60], which is computationally more expensive but exhibits better performance compared to the former.

Despite the countless localization methods that have been developed for various scenarios, all these methods have problems or imperfections of robustly and accurately localizing a robot or a robot team within the indoor environment, where deficient, ambiguous, and repetitive features are usually the case. Though many methods are tested in the indoor environment or the GPS-denied area, the testing environment does not explicitly deal with all the above challenging situations.

In this thesis, two new cooperative localization methods MOMA and S-MOMA, are proposed. MOMA belongs to the first category and extended the idea in [31] by using cheap cameras and printed planar fiducial markers instead of an expensive laser system. Based on MOMA, S-MOMA is developed, which includes environment-interaction to the cooperative MOMA approach. The method is a hybrid of the first and the second category. The method retains the concept of the portable fiducial landmark from the environment-noninteractive framework for robustness considerations. Meanwhile, the algorithm allows the system to interact with its surrounding environment and to be further influenced by it.

3 Eye-to-eye Calibration

In this chapter, the theoretical foundation is first provided, which is necessary to understand the succeeding derivations. A thorough explanation of Liu's method and the inherent instability existing in the method is provided afterward, based on which the weighted non-linear optimization method and the measurement selection strategy are presented. Then a new eye-to-eye calibration method applying a highly accurate tracking system is introduced. The optimization method and the data selection strategy are validated on synthetic data as well as in a real experiment. The proposed method applying the tracking system is also implemented in a real experiment, which serves as the benchmark for other calibration methods and configurations.

3.1 Preliminaries

This section provides mathematical foundations, namely Lie Group, Lie Algebra, and bundle adjustment (BA), which will be frequently used for the subsequent derivations. For simplicity, these definitions introduced in the following subsections are kept plain and concise. A more detailed and comprehensive explanation could be found in [43], [6].

3.1.1 Lie Group and Lie Algebra

One of the most common parameterization for rotation is using the 3D rotation matrix: $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3]$. The rotation matrix has two properties: all of its column vectors or row vectors are orthogonal, and its determinant equals to 1. The construction of non-linear optimization problems such as BA or PnP takes the camera pose as one of their variables. In this case, the derivative of the objective with respect to the camera pose is required. Due to the above inherent constraints, it is not possible to directly optimize variables that are in the rotation matrix form $\mathbf{R}$ or the transformation matrix form $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$, which includes $\mathbf{R}$.

The way to get around the above limitation is to apply the relationship between Lie Group and Lie Algebra, which allows the pose estimation to be transformed into an unconstrained optimization problem.

Lie Group

A group is defined as an algebraic structure that consists of a set of elements together with a binary operation $\otimes$. Any two elements (a, b) in the set S could be combined by the operator to form a third element c. The operation $\otimes$ must meet four group axioms in order to form the group $G = (S, \otimes)$, namely closure, associativity, identity, and invertibility.

- Closure $\forall a, b \in S,\ \ a \otimes b \in S.$
- Associativity $\forall a, b, c \in S,\ \ (a \otimes b) \otimes c = a \otimes (b \otimes c).$
- Identity $\exists e \in S,\ \ s.t.\ \forall a \in S,\ \ e \otimes a = a \otimes e = a.$
- Invertibility $\forall a \in S,\ \ \exists a^{-1} \in S,\ \ s.t.\ a \otimes a^{-1} = e.$

Examples of group are: $G = (\mathbb{Z}, +)$ that is composed of all the integers with the addition operation; Special Orthogonal Group denoted as SO(3) that consists of 3D rotation matrix $\mathbf{R}$ with the multiplication operation; Special Euclidean Group SE(3) consisting of the transformation matrix $\mathbf{T}$ under the multiplication operation.

$$\mathrm{SO(3)} = \left\{ \mathbf{R} \in \mathbb{R}^{3\times 3} \mid \mathbf{R}\mathbf{R}^{\mathrm{T}} = \mathbf{I}, \left|\mathbf{R}\right| = 1 \right\}.$$

$$\mathrm{SE(3)} = \left\{ \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} \mid \mathbf{R} \in \mathrm{SO(3)}, \mathbf{t} \in \mathbb{R}^{3} \right\}.$$

Different from the general groups whose elements could be discrete, Lie group is a group as well as a differentiable manifold, on which the operation $\otimes$ is a smooth map. Since both SO(3) and SE(3) have a natural structure as a manifold and the group operations are smooth, they are both Lie groups.

Lie Algebra

A Lie algebra consists of a vector space $\mathbb{V}$ over some field $\mathbb{F}$ and a non-associative binary operation $[,]$, which is called Lie bracket.

A Lie algebra ($\mathbb{V}$, $\mathbb{F}$, $[,]$) should satisfy the following axioms:

- Closure $\forall \mathbf{C}, \mathbf{D} \in \mathbb{V},\ \ [\mathbf{C}, \mathbf{D}] \in \mathbb{V}.$
- Bilinearity $\forall \mathbf{C}, \mathbf{D}, \mathbf{Z} \in \mathbb{V},\ \ a, b \in \mathbb{F}$
  $[a\mathbf{C} + b\mathbf{D}, \mathbf{Z}] = a[\mathbf{C}, \mathbf{Z}] + b[\mathbf{D}, \mathbf{Z}],\ \ [\mathbf{Z}, a\mathbf{C} + b\mathbf{D}] = a[\mathbf{Z}, \mathbf{C}] + b[\mathbf{Z}, \mathbf{D}].$

- Alternativity $\forall \mathbf{C} \in \mathbb{V}, \quad [\mathbf{C},\mathbf{C}] = 0$.
- The Jacobian identity $\forall \mathbf{C},\mathbf{D},\mathbf{Z} \in \mathbb{V}$, $[\mathbf{C},[\mathbf{D},\mathbf{Z}]]+[\mathbf{Z},[\mathbf{C},\mathbf{D}]]+[\mathbf{D},[\mathbf{Z},\mathbf{C}]] = 0$.
- Anticommutativity $\forall \mathbf{C},\mathbf{D} \in \mathbb{V}, \quad [\mathbf{C},\mathbf{D}] = -[\mathbf{D},\mathbf{C}]$.

Each Lie group has its corresponding Lie algebra. The Lie algebra of SO(3) is denoted as $\mathfrak{so}(3)$ consisting of the vectors $\boldsymbol{\phi}$ in $\mathbb{R}^3$. Each $\boldsymbol{\phi}$ could generate the corresponding skew-symmetric matrix Φ as follows after applying the operator '$\wedge$':

$$\Phi = \boldsymbol{\phi}^\wedge = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}.$$

The Lie bracket is defined as $[\boldsymbol{\phi}_1,\boldsymbol{\phi}_2] = (\Phi_1\Phi_2 - \Phi_2\Phi_1)^\vee$, where the operator '$\vee$' transforms a skew-symmetric matrix into its corresponding vector form. So the $\mathfrak{so}(3)$ could then be represented as:

$$\mathfrak{so}(3) = \left\{\boldsymbol{\phi} \in \mathbb{R}^3, \boldsymbol{\Phi} = \boldsymbol{\phi}^\wedge \in \mathbb{R}^{3\times 3}\right\}.$$

$\mathfrak{se}(3)$ consisting of the vectors $\boldsymbol{\rho}$ in $\mathbb{R}^6$ represents the Lie algebra of SE(3), and is denoted as follows:

$$\mathfrak{se}(3) = \left\{\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix} \in \mathbb{R}^6, \boldsymbol{\rho} \in \mathbb{R}^3, \boldsymbol{\phi} \in \mathfrak{so}(3)\right\}.$$

The vector $\boldsymbol{\rho}$ in $\boldsymbol{\xi}$ represents the translational part, and $\boldsymbol{\phi}$ describes the rotational part. Similar to $\mathfrak{so}(3)$, the operator '$\wedge$' transforms a vector in $\mathbb{R}^6$ into a 4×4 matrix, except that the obtained matrix is no longer skew-symmetric like in $\mathfrak{so}(3)$.

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho} \\ \mathbf{0}^\mathrm{T} & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4}.$$

The Lie bracket of $\mathfrak{se}(3)$ is defined as:

$$[\boldsymbol{\xi}_1,\boldsymbol{\xi}_2] = (\boldsymbol{\xi}_1^\wedge\boldsymbol{\xi}_2^\wedge, \boldsymbol{\xi}_2^\wedge\boldsymbol{\xi}_1^\wedge)^\vee.$$

Exponential Mapping

The vector $\boldsymbol{\phi}$ of Lie algebra $\mathfrak{so}(3)$ could be represented as a unit vector $\boldsymbol{a}$ describing the axis of the rotation about which an angle θ is rotated according to the right-hand rule. Then the rotation matrix $\mathbf{R}$ is related to the elements in $\mathfrak{so}(3)$ by the following exponential mapping:

$$\mathbf{R} = \exp(\boldsymbol{\phi}^{\wedge}) = \exp(\theta \boldsymbol{a}^{\wedge}) = \sum_{n=0}^{\infty} \frac{1}{n!}(\theta \boldsymbol{a}^{\wedge})^{n} = cos(\theta \mathbf{I}) + (1 - cos\theta))\boldsymbol{a}\boldsymbol{a}^{\mathrm{T}} + sin(\theta \boldsymbol{a}),$$

where $\boldsymbol{\phi}_j^{\wedge}$ represents the corresponding skew-symmetric matrix, and 'exp()' in the above equation defines the exponential map from $\mathfrak{so}(3)$ to the Special Orthogonal Group SO(3) [67]. The result is the same as the Rodrigues' rotation formula [7].

The following exponential mapping relates the transformation matrix $\mathbf{T}$ to the elements in $\mathfrak{se}(3)$:

$$\mathbf{T} = \exp(\boldsymbol{\xi}^{\wedge}) = \begin{bmatrix} \sum_{n=0}^{\infty} \frac{1}{n!}(\boldsymbol{\phi}^{\wedge})^{n} & \sum_{n=0}^{\infty} \frac{1}{(n+1)!}(\boldsymbol{\phi}^{\wedge})^{n}\boldsymbol{\rho} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \boldsymbol{J}\boldsymbol{\rho} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix},$$

where

$$\boldsymbol{J} = \frac{sin\theta}{\theta}\mathbf{I} + (1 - \frac{sin\theta}{\theta})\boldsymbol{a}\boldsymbol{a}^{\mathrm{T}} + \frac{1 - cos\theta}{\theta}\boldsymbol{a}^{\wedge}.$$

After the exponential mapping, the operator '$\wedge$' transforms $\boldsymbol{\xi}$ to the transformation matrix $\mathbf{T}$. The translational vector $\mathbf{t}$ of $\mathbf{T}$ now becomes the product of the linear transformation $\boldsymbol{J}$ and the vector $\boldsymbol{\rho}$ in $\mathfrak{se}(3)$. The transformation $\boldsymbol{J}$ is only related to rotation. For more details, we refer to [67].

3.1.2 Bundle Adjustment

Perspective-*n*-Point (P*n*P) estimates the pose of the camera when the initially estimated position of a set of 3D spatial feature points and their 2D projections are given. In the VO and SLAM framework where a stereo camera or an RGB-D camera is used, P*n*P could be directly applied for pose estimation. There are different solutions to the P*n*P problem, such as P3P [21], direct linear transformation (DLT) [25], efficient P*n*P (EP*n*P) [35], robust pose estimation by actively controlling planar point configurations [3], BA, etc. Of all these P*n*P methods, BA generates the most accurate estimation.

Different from P3P, DLT, and EP*n*P methods, where the camera pose is first estimated and the 3D feature position is then calculated, BA simultaneously refines all the parameters, including the 3D coordinates of the features and the camera pose $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$. Given a number of n initially estimated 3D features

$\mathbf{P}_i = [X_i, Y_i, Z_i, 1]^{\mathrm{T}}$, $i \in [1, n]$ represented in the world frame, and their corresponding 2D projections on the image $\mathbf{p}_i = [u_i, v_i, 1]^{\mathrm{T}}$, they are related by the pinhole camera model:

$$\lambda_i \mathbf{p}_i = \mathbf{K}\,\mathbf{T}\mathbf{P}_i,$$

where λ_i is the depth of the feature point and $\mathbf{K} \in \mathbb{R}^{3\times 3}$ is the intrinsic camera calibration matrix.

Due to the image noise, the above equation does not hold. The objective function of BA is built on this error item which is formulated as:

$$(\hat{\mathbf{T}}, \hat{\mathbf{P}}_1, \hat{\mathbf{P}}_2 \cdots \hat{\mathbf{P}}_n) = \underset{\mathbf{T},\mathbf{P}_1,\mathbf{P}_2\cdots\mathbf{P}_n}{\arg\min} \sum_{i=1}^{i=n} \left\| \mathbf{p}_i - \frac{1}{\lambda_i}\mathbf{K}\,(\mathbf{T}\mathbf{P}_i) \right\|_2^2 = \underset{\mathbf{T},\mathbf{P}_1,\mathbf{P}_2\cdots\mathbf{P}_n}{\arg\min} \sum_{i=1}^{i=n} \|\boldsymbol{\varepsilon}_i\|_2^2.$$

The objective function is a typical non-linear least-squares problem. Each error item (residual) $\boldsymbol{\varepsilon}_i$ in the equation is known as the reprojection error, which depicts the difference between the real measurement and the predicted projection, which is based on the currently estimated 3D feature position and the camera pose. By minimizing the overall projection error, the camera pose, and the 3D feature position are estimated to their optimum.

3.2 Data Selection Strategy and Weighted Optimization Method

As mentioned before, the optimization method and the measurement selection strategy are suitable for the setup that builds its objective function based on the reprojection error of 3D-2D point correspondences constrained by 3D-3D closed-loop pose transformation $\mathbf{AX} = \mathbf{YB}$. In this section, the weighted non-linear optimization method and the data selection strategy are presented using the example of Liu's setup [39].

3.2.1 Liu's Method

In order to understand the underlying instabilities in Liu's method, the method is first explained.

Liu's method uses a mobile calibration device which rigidly links two planar calibration patterns *P1* and *P2* whose relative pose $\mathbf{X}$ is unknown (Figure3.1). The planar calibration pattern could be a fiducial marker or a chessboard. By changing the pose of the calibration rig relative to the camera, a set of images $\{\mathbf{I}_i^{P1}, \mathbf{I}_i^{P2}\}_{i=1}^{i=n}$ containing the calibration patterns is collected, based on which

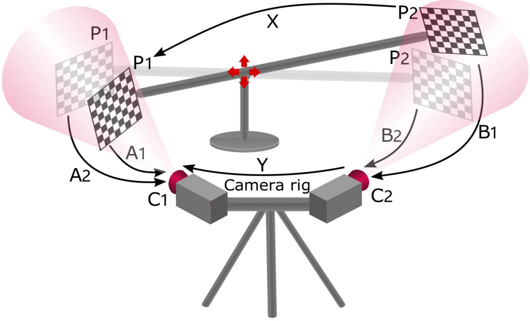


Figure 3.1: The measurement collection process of Liu's setup.

$\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ could be recovered. The initial estimation of $\mathbf{X}$ and $\mathbf{Y}$ is calculated by solving $\mathbf{AX} = \mathbf{YB}$. In order to further improve the calibration accuracy, the initial value of $\mathbf{X}$ and $\mathbf{Y}$ is then applied to minimize the objective function, which is based on the reprojection error from all measurements.

Since two pattern detection processes coexist in this setup and each recovered relative pose pair $(\mathbf{A}_i, \mathbf{B}_i)$ is restricted by the same constraint $\mathbf{A}_i\mathbf{X} = \mathbf{YB}_i$, in what follows, many dual equations will be derived. For simplicity, the explanation of the derivation will focus on one side with the calibration pattern *P1*, while the conclusions for the other pattern *P2* are given without explicit explanation.

The classical minimization of the reprojection error between 3D marker points and their corresponding projections is first summarized using the notations that appear in Liu's calibration setup. Given a 3D-2D point correspondence of j-th 3D marker point with coordinates $\mathbf{M}_j^{P1} = [X_j^{P1}, Y_j^{P1}, Z_j^{P1}]^{\mathrm{T}} \in \mathbb{R}^3$ represented in the fiducial pattern frame *P1* and its corresponding projection onto a calibrated camera[1] with image coordinates $\mathbf{m}_j^{P1} = [x_j^{P1}, y_j^{P1}]^{\mathrm{T}} \in \mathbb{R}^2$, the relationship between these points is given by the relative pose $g = (\mathbf{R}_A, \mathbf{t}_A)$ (Euclidean transformation) between the pattern frame *P1* and the camera frame *C1*, $\mathbf{M}_j^{C1} = \mathbf{R}_A \mathbf{M}_j^{P1} + \mathbf{t}_A$, followed by a projection π with $\mathbf{m}_j^{P1} = \pi(\mathbf{M}_j^{C1}) = [X_j^{C1}/Z_j^{C1}, Y_j^{C1}/Z_j^{C1}]^{\mathrm{T}}$. This leads to the relation:

$$\mathbf{m}_j^{P1} = \pi(\mathbf{M}_j^{C1}) = \pi(\mathbf{R}_A \mathbf{M}_j^{P1} + \mathbf{t}_A). \tag{3.1}$$

The projection process of the pattern *P2* in the camera frame *C2* is expressed

[1] Assuming the calibration matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$ to be known, the homogeneous image coordinates in pixel $\overline{\mathbf{m}}'_j = [x'_j, y'_j, 1]^{\mathrm{T}}$ can be transformed to homogeneous normalized image coordinates in metric units $\overline{\mathbf{m}}_j = \mathbf{K}^{-1}\overline{\mathbf{m}}'_j$.

as follows:

$$\mathbf{m}_l^{P2} = \pi(\mathbf{M}_l^{C2}) = \pi(\mathbf{R}_B \mathbf{M}_l^{P2} + \mathbf{t}_B).$$

In the real experiment, the above projection procedure will be corrupted due to the undesirable electronic noise. Including additive noise $\boldsymbol{\varepsilon}_j^{P1} = [\varepsilon_j^{P1}, \zeta_j^{P1}]^{\mathrm{T}}$ to the error-free image coordinates $\mathbf{m}_j^{P1}$, the noisy measurements of the image coordinates could be represented as $\tilde{\mathbf{m}}_j^{P1} = \mathbf{m}_j^{P1} + \boldsymbol{\varepsilon}_j^{P1}$. Thus, the reprojection error $\|\boldsymbol{\varepsilon}_j^{P1}\|_2^2 = \|\tilde{\mathbf{m}}_j^{P1} - \mathbf{m}_j^{P1}\|_2^2$ of each point could be solved, which is a squared 2-norm. Minimizing the squared 2-norm of all points for the optimal pose $(\hat{\mathbf{R}}_A, \hat{\mathbf{t}}_A)$ leads to the following least-squares estimator:

$$(\hat{\mathbf{R}}_A, \hat{\mathbf{t}}_A) = \arg\min_{\mathbf{R}_A, \mathbf{t}_A} \sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_j^{P1}\|_2^2, \quad m \geq 3. \tag{3.2}$$

A similar equation can be drawn for pattern *P2*:

$$(\hat{\mathbf{R}}_B, \hat{\mathbf{t}}_B) = \arg\min_{\mathbf{R}_B, \mathbf{t}_B} \sum_{l=1}^{o} \|\boldsymbol{\varepsilon}_l^{P2}\|_2^2, \quad o \geq 3. \tag{3.3}$$

The formulation of Liu's method is similar to the hand-eye robot-world calibration routine, which uses a number of n pose pair measurements $\{\mathbf{A}_i\}_{i=1}^{i=n}$ and $\{\mathbf{B}_i\}_{i=1}^{i=n}$, where $\mathbf{A}_i$ is the transform from the world frame to the camera frame and $\mathbf{B}_i$ denotes the relationship between the robot-base and the robot-hand frame. In Liu's method, the above pose pair measurements are replaced by a set of marker-eye poses $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$. Applying the 3D closed-loop pose constraints for $\mathbf{A}_i$ generates the following equations:

$$\begin{aligned} \mathbf{A}_i &= \mathbf{Y}\mathbf{B}_i\mathbf{X}^{-1}, \\ \mathbf{R}_{A_i} &= \mathbf{R}_Y \mathbf{R}_{B_i} \mathbf{R}_X^{\mathrm{T}}, \qquad (3.4) \\ \mathbf{t}_{A_i} &= \mathbf{R}_Y(\mathbf{t}_{B_i} - \mathbf{R}_{B_i}\mathbf{R}_X^{\mathrm{T}}\mathbf{t}_X) + \mathbf{t}_Y. \qquad (3.5) \end{aligned}$$

The above counterpart for $\mathbf{B}_i$ applying the same 3D closed-loop pose constraints is:

$$\begin{aligned} \mathbf{B}_i &= \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}, \\ \mathbf{R}_{B_i} &= \mathbf{R}_Y^{\mathrm{T}} \mathbf{R}_{A_i} \mathbf{R}_X, \qquad (3.6) \\ \mathbf{t}_{B_i} &= \mathbf{R}_Y^{\mathrm{T}}(\mathbf{R}_{A_i}\mathbf{t}_X + \mathbf{t}_{A_i} - \mathbf{t}_Y). \qquad (3.7) \end{aligned}$$

Combining all those corresponding 3D pose constraints with the minimization of the reprojection error (3.2) by including the constraints (3.4) and (3.5) into (3.1) gives

$$\mathbf{m}_{ij}^{P1} = \pi\left(\mathbf{R}_Y\left(\mathbf{R}_{B_i}\mathbf{R}_X^{\mathrm{T}}(\mathbf{M}_j^{P1} - \mathbf{t}_X) + \mathbf{t}_{B_i}\right) + \mathbf{t}_Y\right), \tag{3.8}$$

which leads to the optimization similarly formulated in [64]

$$(\hat{\mathbf{R}}_X, \hat{\mathbf{t}}_X, \hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_X, \mathbf{t}_X, \mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n}\sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2.$$

Here, the error term extends to $\|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2 = \|\tilde{\mathbf{m}}_{ij}^{P1} - \mathbf{m}_{ij}^{P1}\|_2^2$.

In Liu's setup, each projection $\mathbf{m}_{il}^{P2}$ can also be constrained by all corresponding 3D closed-loop pose equations $\mathbf{AX} = \mathbf{YB}$. Solving the constraints for $\mathbf{B}_i$ gives the following: for each pose pair configuration i, another reprojection error $\|\boldsymbol{\varepsilon}_{il}^{P2}\|_2^2 = \|\tilde{\mathbf{m}}_{il}^{P2} - \mathbf{m}_{il}^{P2}\|_2^2$ could be obtained. Including constraints (3.6) and (3.7) into the corresponding projection that is part of (3.3) leads to an equation similar to (3.8):

$$\mathbf{m}_{il}^{P2} = \pi\left(\mathbf{R}_Y^{\mathrm{T}}\left(\mathbf{R}_{A_i}(\mathbf{R}_X\mathbf{M}_l^{P2} + \mathbf{t}_X) + \mathbf{t}_{A_i} - \mathbf{t}_Y\right)\right).$$

Based on the combination of all these additional constrained projections, the extended optimization problem can be formulated as follows:

$$(\hat{\mathbf{R}}_X, \hat{\mathbf{t}}_X, \hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_X, \mathbf{t}_X, \mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n}(\sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2 + \sum_{l=1}^{o} \|\boldsymbol{\varepsilon}_{il}^{P2}\|_2^2). \tag{3.9}$$

In general, the number of point correspondences m for the first camera *C1* can differ in the number of point correspondences o for the second camera *C2*. This optimization is similar to BA for two cameras. The difference is that the two different bundles of rays for camera *C1* and *C2* do not belong to the same 3D points but different 3D points on two different calibration patterns. In this case, each marker is only visible in the FOV of only one camera. Therefore, an additional constraint that relates to the 3D points of each bundle is needed, which is given by $\mathbf{AX} = \mathbf{YB}$ originally used for hand-eye and robot-world calibration.

So far, the derived objective function (3.9) is the same as the one optimized in [39]. It is non-convex, so the iterative optimization can only guarantee to converge to a local minimum, and a proper initialization is needed in order to reach a good estimation. Typical methods for solving $\mathbf{AX} = \mathbf{YB}$ could be found in [17] [36] [56] [68]. In this thesis, the initial value of $\mathbf{X}$ and $\mathbf{Y}$ is calculated by applying the method in [68] beforehand.

3.2.2 Underlying Instabilities in Liu's Method

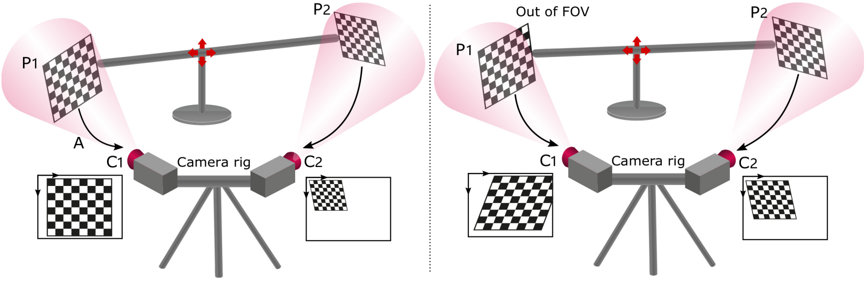


Figure 3.2: An example showing the twisted change in the resulting images after a minor pose adjustment. Considering that the measurement quality of the pattern *P2* in the left figure is worse, the camera rig moves a little in the neighborhood of the current pose ($\mathbf{A}_1$, $\mathbf{B}_1$) in order to improve its quality. After the movement, even though the quality of the pattern *P2* gets better, the calibration pattern *P1* is no longer entirely in the FOV of the camera *C1*, which makes the pose pair ($\mathbf{A}_2$, $\mathbf{B}_2$) in the right figure invalid.

The measurement quality refers to the resolution of the captured calibration pattern. The measurement space or the pose change space is defined as a collection of pose pairs $\mathbf{A}_i$ and $\mathbf{B}_i$, which are the relative pose between the camera *C1*, *C2*, and the corresponding calibration pattern *P1*, *P2*. All the pose pairs in the measurement space meet the following conditions. First, for each pose pair, both planar calibration patterns have to be in the FOV of the corresponding cameras such that all the coordinates of the projections $\{\tilde{\mathbf{m}}_{ij}^{P1}\}_{j=1}^{m}$ and $\{\tilde{\mathbf{m}}_{il}^{P2}\}_{l=1}^{o}$ could be extracted without outliers. Second, the resulting measurement quality from each pose pair should be above a certain threshold since higher resolution indicates a better estimation of $\mathbf{A}_i$ and $\mathbf{B}_i$. As mentioned in the first chapter, Liu's method is not always stable due to the **reduced**, **twisted** pose change space and the **imbalanced** measurement quality [71].

One primary practical issue to reach accurate calibration results is a proper set of accurately estimated measurement pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ covering all six degrees of freedom of the pose $\mathbf{X}$ and $\mathbf{Y}$. However, with the assistance of the customized calibration device, collecting such a set of measurement pairs is problematic because of the rigid coupling between the two patterns, whose effect is disastrous.

First and most straightforward, the closed-loop coupling reduces the pose change space since both of the calibration patterns have to appear within the FOV of the corresponding camera. The second consequence resulting from the coupling is the twisted pose change space. Due to the coupling $\mathbf{B}_i = \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}$, a minor change in pose $\mathbf{A}_i$ would lead to a compound change in $\mathbf{B}_i$. The same happens with a minor change in pose $\mathbf{B}_i$. The twisted pose change space indicates the hardness of capturing both calibration patterns with high resolution (Figure 3.2). After a minor change in the pose of the calibration rig, the calibration pattern $P1$ is not completely covered by the FOV of the camera $C1$. Besides, this property also adds another layer of difficulty to the data collection process making it quite anti-intuitive. Though some pose pairs with good quality are theoretically valid, they are challenging to acquire in reality. What is more, the twisted pose change exists in every pose pair, which means pose pairs that are spatially close to each other might result in very different images.

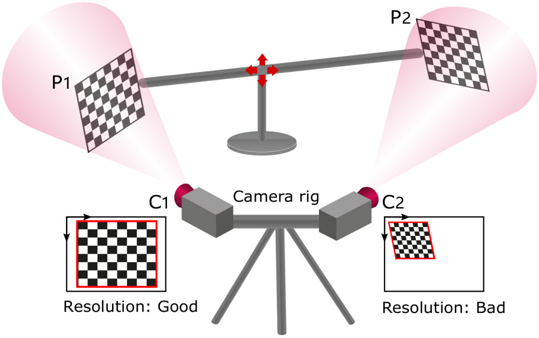


Figure 3.3: Relationship between the pose of the calibration rig relative to the camera pair and the corresponding resolution quality. When the calibration pattern from one side of the rig is placed near to the camera, an image with high resolution will be captured. In contrast, the calibration pattern from the other side will be captured with a comparatively lower resolution and vice versa. Corresponding images containing different fiducial patterns are captured at the same measurement time, which are of different projection sizes. The projection size is defined as the area surrounded by the red lines on each image. In this example, the pattern *P1* generates a larger projection size on the image, hence higher resolution than *P2*. When both images are corrupted by the same level of noise, the pose estimation using the measurement of *P1* will be less sensitive to noise and will produce a better pose estimation.

Another adverse effect caused by the closed-loop constraint is the imbalanced measurement quality. Because of the coupling, the placement of one calibration pattern will influence the placement of the other one. Hence, acquiring a set of images that are of high resolution for both calibration patterns is very challenging. Figure 3.3 demonstrates the relationship between the captured measurement quality and the relative pose between the calibration rig and the camera pair. Meanwhile, explicit images are presented to explain further this problem, which gives an imbalanced projection size of different calibration patterns for the same pose pair. Here, the projection size is used as the indicator of the measurement quality. Larger projection size indicates better measurement quality.

From the perspective of robustness, images of the calibration patterns should be taken from as many different poses as possible. While for accuracy consideration, the calibration objects should be captured with as much resolution as possible since the high resolution of the pattern results in better pose estimation between the camera and the calibration object. Generally, during the collection of the measurements, balances have to be kept between the pose pair variety and the measurement quality.

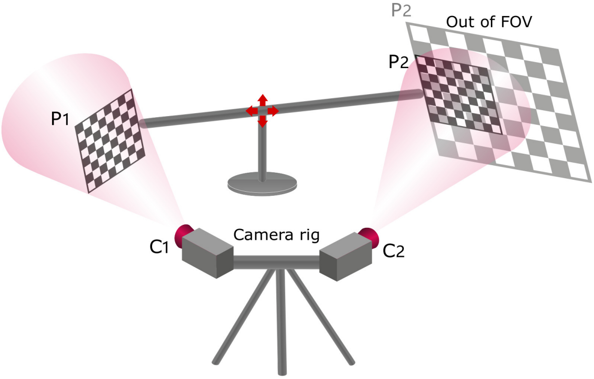


Figure 3.4: The demonstration of the influence of $\mathbf{X}$ on the pattern size. In this example, two different sizes are provided for the calibration pattern *P2* with the center points of the left side overlapping. Compared to the size of the pattern *P1*, the calibration pattern *P2* with a larger size is impossible to generate a decent measurement space.

In the end, a discussion of two variables which influence the measurement space and the measurement quality is given. These two variables are the size of the calibration pattern and the relative pose $\mathbf{X}$ between the calibration objects. It is not possible to capture the whole calibration pattern that has a huge size (Figure 3.4), while the calibration pattern with a tiny size could not be captured with good quality. Therefore, there must exist at least one optimal calibration pattern size, which generates a larger measurement space. The same happens with different choices of $\mathbf{X}$. With some poses $\mathbf{X}$, it is easier to generate larger measurement space compared to the others. In Figure 3.5, two different values of $\mathbf{X}$ are chosen. In this example, the choice of $\mathbf{X}_1$ is better than $\mathbf{X}_2$ since the former allows more valid pose pairs to be collected. The tricky questions is: is $\mathbf{X}_1$ the most optimal choice? If not, how to find the most optimal $\mathbf{X}$?

Moreover, these two variables are not independent. When the size of the calibration pattern is determined, the relative pose $\mathbf{X}$ should be coordinated in order

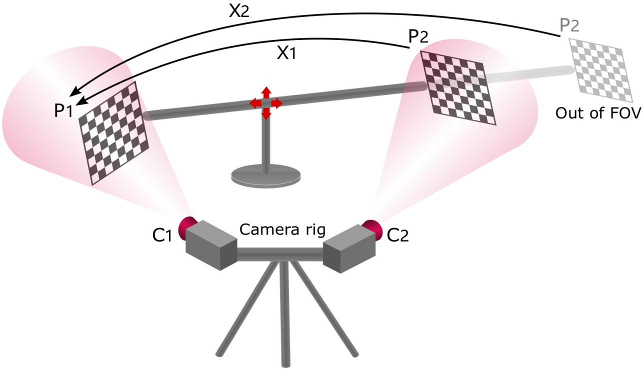


Figure 3.5: Different values of **X** will create different pose change space. In the example, the translation of $\mathbf{X}_2$ is much wider than that of the camera pair, which makes it harder to generate appropriate pose pairs compared to $\mathbf{X}_1$.

to fit the pattern size better. It is true the other way around. The size of the pattern should also be accordingly adjusted once the relative pose **X** is fixed such that a larger measurement space with better measurement quality could be guaranteed. In this thesis, the influence of these two variables remains an open question, and the relevant study is not explored.

The calibration rig used in Liu's setup has to be particularly manufactured, so an estimated calibration pattern size, as well as the relative pose **X** are needed before the construction of the rig. Based on the above analysis, their values are not theoretically evident and could be better determined or re-adjusted from the real experiment. In this case, commonly, roughly estimated values are first tested and rectified, which brings extra complexity and inconvenience to the calibration procedure.

Based on the above analysis, it is necessary to stress that although it is simple to collect pose pairs applying Liu's setup, particular attention is needed in order to generate a pose pair set with good quality and comparatively scattered spatial distribution, which are essential for accurate calibration. However, with so many undecided, intertwined variables and inherent hindrances, it is both theoretically and practically impossible to provide a valid paradigm for collecting an optimal set of pose pair. So instead of excessively focusing on the challenging data collection procedure, the weighted non-linear optimization method, together with the data selection strategy that is going to be presented, aims to solve the following question, which is more meaningful and practical. If provided with a collected measurement set, how to generate the most possibly accurate and robust calibration results, which are less sensitive to the gathered measurements?

3.2.3 Data Selection Strategy

The reprojection error based objective function needs an initial value of $\mathbf{X}$ and $\mathbf{Y}$, which is estimated from solving $\mathbf{AX} = \mathbf{YB}$. The methods of solving the equation demand a set of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ covering the six degrees of freedom.

The methods for solving $\mathbf{AX} = \mathbf{YB}$ are mostly aimed at the hand-eye, robot-world calibration problem. The camera mounted on the robot arm is placed at different poses to capture the calibration pattern. In the hand-eye, robot-world calibration problem, the measurement space depends on the following factors: the movement space of the robot arm, the transform $\mathbf{X}$ from the robot base to the calibration pattern (the world frame), the FOV of the camera and the size of the calibration pattern. The first factor indicates the movable range $\mathbb{S}1$ of the camera, while the last three factors determine the space range $\mathbb{S}2$, within which the camera could capture the calibration pattern. Thus, the measurement space $\mathbb{S}$ is the intersection of the space $\mathbb{S}1$ and $\mathbb{S}2$: $\mathbb{S} = \mathbb{S}1 \cap \mathbb{S}2$. In order to obtain the possibly largest measurement space, the space $\mathbb{S}2$ should overlap as much as possible with $\mathbb{S}1$ since the latter is hardware-dependent and could not be changed. In contrast, the former could be adjusted by regulating the robot-world pose $\mathbf{X}$ and the calibration pattern size. In most cases, the movement space of the robot arm $\mathbb{S}1$ is large enough, and an abundant number of different pose pairs could be generated within the resulting measurement space $\mathbb{S}$.

The pose change space in Liu's setup is complicated. As explained before, the pose change space in Liu's setup is reduced due to the closed-loop coupling, which would decrease the accuracy of solving $\mathbf{AX} = \mathbf{YB}$ if the spatial distribution of the collected pose pairs is not scattered enough. Besides, a minor change in the pose pair results in a compound, twisted change in the resulting images, which makes it less straightforward to distinguish the spatial difference between different pose pairs. Pose pairs that are spatially close might result in very different images. Thus, collected pose pairs need to be carefully handled. Otherwise, they would bring potential hazards to calibration stability.

On the other hand, the estimation accuracy of the pose pair is also crucial to the solution of $\mathbf{AX} = \mathbf{YB}$ since better estimated pose pairs improve the initial estimation of $\mathbf{X}$ and $\mathbf{Y}$. Therefore, the measurement quality should work in conjunction with the measurement space in order to generate an optimal initial value of $\mathbf{X}$ and $\mathbf{Y}$.

Normally, a large number of pose pairs are suggested during the data collection process to cover as much measurement space as possible. An extra data selection filter is then applied to all the collected pose pairs. The filter calculates the rotational difference e^R and translational difference e^T between all the pose pairs based on the criteria formulated in (3.16), (3.17), as well as the projection size S^P of the calibration pattern. The pose pair $(\mathbf{A}_i, \mathbf{B}_i)$ whose rotational difference or translational difference are below certain threshold θ^R, γ^T compared to all the rest pose pairs $(\mathbf{A}_j, \mathbf{B}_j)(j \neq i)$ will be excluded: $(e^R(\mathbf{A}_i, \mathbf{A}_j) < \theta^R \parallel e^R(\mathbf{B}_i, \mathbf{B}_j) < \theta^R \parallel e^T(\mathbf{A}_i, \mathbf{A}_j) < \gamma^T \parallel e^T(\mathbf{B}_i, \mathbf{B}_j) < \gamma^T)$. Besides, the pose pairs whose resulting projection size of all the captured calibration pattern is smaller than some pre-defined value τ will not be included: $(S_i^{P1} < \tau \parallel S_i^{P2} < \tau)$.

In the end, a subset of the collected pose pair, which is chosen with more discretion, is used to solve $\mathbf{AX} = \mathbf{YB}$, which provides a better estimated initial value of $\mathbf{X}$ and $\mathbf{Y}$ for the following non-linear optimization.

3.2.4 Weighted Non-linear Optimization Method

The underlying measurement imbalance discourages the objective function from including all the measurements and treating them equally. Considering the unpleasant imbalance of the measurement quality which leads to the diversity of the projection size of different calibration patterns within one measurement pair, additional weightings λ_i^{P1} and λ_i^{P2} are introduced to the objective (3.9), which leads to:

$$(\hat{\mathbf{R}}_X, \hat{\mathbf{t}}_X, \hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_X, \mathbf{t}_X, \mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\lambda_i^{P1} \sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2 + \lambda_i^{P2} \sum_{l=1}^{o} \|\boldsymbol{\varepsilon}_{il}^{P2}\|_2^2). \tag{3.10}$$

The weighting λ_i^{P1} used for the reprojection error related to the pattern *P1* is chosen to be the square root of the projection size S_i^{P2} of the pattern *P2* normalized by the full image size S_{max}:

$$\lambda_i^{P1} = \sqrt{S_i^{P2} / S_{max}}.$$

The other weighting factor λ_i^{P2} is calculated in a similar way:

$$\lambda_i^{P2} = \sqrt{S_i^{P1} / S_{max}}.$$

The reason for choosing such weighting lies in the replacement of $\mathbf{A}_i$ with $\mathbf{YB}_i\mathbf{X}^{-1}$. The reprojection error produced from the calibration pattern *P1* now depends on its replacement $\mathbf{YB}_i\mathbf{X}^{-1}$ which has the pose estimation $\mathbf{B}_i$ inside, so the estimation accuracy of $\mathbf{B}_i$ influences the reprojection error of *P1*: $\mathbf{B}_i$ with better estimation quality should have more influence on the optimization results, and this leads to a higher weight λ_i^{P1}. In this case, the projection size is regarded as an indirect indicator of the measurement quality. The same happens with the replacement of $\mathbf{B}_i$.

Though the method is explained using Liu's setup, the proposed optimization method could be applied to different setups. To be more specific, the method could be applied to the setups, which minimize the sum of reprojection errors and are constrained by the closed-loop pose transformations $\mathbf{AX} = \mathbf{YB}$. The integration of the method improves calibration accuracy and stability. The introduced weighting factor allows more pose pairs to be safely included in the calibration procedure since their influence on the estimation is now correlated with the quality indicator, namely the projection size of the captured calibration pattern.

3.3 Eye-to-eye Calibration Applying Highly Accurate Tracking System

One distinct advantage of Liu's setup is that the introduced fiducial features are accurately known a prior, which does not contain any error compared to the map-building based setups. The fiducial features could be included in the non-linear refinement to improve the estimation accuracy further. However, Liu's method is not stable due to the reduced, twisted measurement space, and the imbalanced measurement quality. Though the integration of the data selection strategy and the weighted optimization method relieves the instability, the inherent limitations resulting from Liu's setup still bring a negative effect on the calibration results. This section introduces new calibration methods applying a highly accurate tracking system. The introduction of the tracking system into Liu's setup makes the best use of the tracking system while at the same time keeps the advantages brought by the fiducial features.

Two different configurations are feasible depending on whether the calibration objects are linked or not: fixed trackable pattern setup and unfixed trackable pattern setup.

3.3.1 Fixed Trackable Pattern Setup

The fixed trackable pattern setup demonstrated in Figure 3.6 is similar to Liu's setup except that a highly accurate tracking system is integrated, and the tracking targets are attached to the calibration pattern boards *P1* and *P2*. With the assistance of the tracking system, the relative pose $\mathbf{X}$ between the calibration objects, which is unknown in Liu's setup, could be accurately recovered after aligning

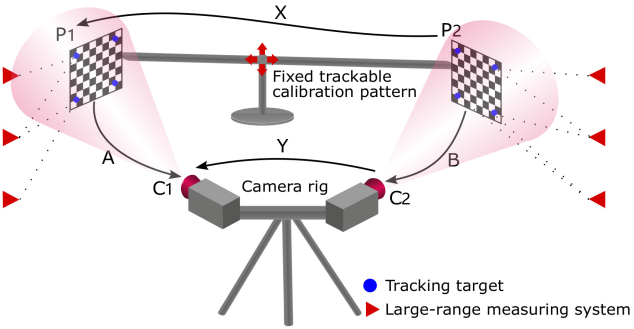

X
P1
P2
Fixed trackable calibration pattern
Y
A
B
C1
Camera rig
C2
● Tracking target
▶ Large-range measuring system

Figure 3.6: The fixed trackable pattern setup applying a highly accurate tracking system.

the tracking targets to the calibration pattern boards in this case. Therefore, the relative pose $\mathbf{Y}$ between the camera pair is the only unknown variable that needs to be estimated.

The data collection procedure is similar to Liu's setup. The calibration rig is placed in several different positions with respect to the camera pair, and a set of images $\{\mathbf{I}_i^{P1}, \mathbf{I}_i^{P2}\}_{i=1}^{i=n}$ with the corresponding calibration patterns as well as the recovered $\mathbf{X}$ from applying the tracking system are gathered. The pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ are then recovered and used to run a final BA (3.11), in which only $\mathbf{Y}$ is optimized:

$$(\hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \arg\min_{\mathbf{R}_Y, \mathbf{t}_Y} \sum_{i=1}^{n} (\sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2 + \sum_{l=1}^{o} \|\boldsymbol{\varepsilon}_{il}^{P2}\|_2^2). \tag{3.11}$$

Based on the closed-loop pose constraint $\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i$, the replacement of $\mathbf{A}_i$ and $\mathbf{B}_i$ becomes:

$$\begin{aligned} \mathbf{A}_i &= \mathbf{Y}\mathbf{B}_i\mathbf{X}^{-1}, \\ \mathbf{B}_i &= \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}, \end{aligned}$$

in which $\mathbf{X}$ is obtained from the tracking system.

Compared to Liu's setup, the fixed trackable pattern setup reduces the number of variables by providing an accurately recovered $\mathbf{X}$ using the tracking system. However, the limited, twisted pose change space and the measurement imbalance have not been eliminated due to the rigid link between the calibration patterns. Therefore, it is necessary to include the weighting factors λ_i^{P1} and λ_i^{P2} similar to (3.10) in order to relieve the above limitations, which leads to the following objective function:

$$(\hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\lambda_i^{P1} \sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_{ij}^{P1}\|_2^2 + \lambda_i^{P2} \sum_{l=1}^{o} \|\boldsymbol{\varepsilon}_{il}^{P2}\|_2^2).$$

3.3.2 Unfixed Trackable Pattern Setup

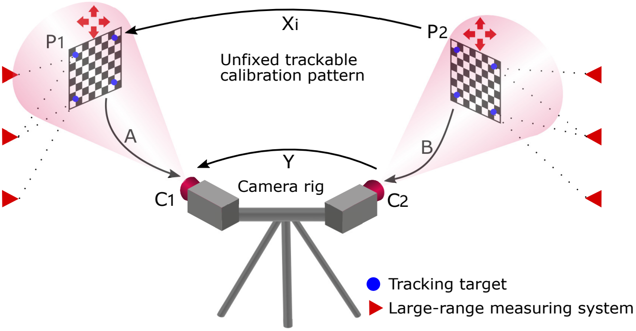

Xi
P1
P2
Unfixed trackable calibration pattern
A
B
Y
Camera rig
C1
C2
● Tracking target
▶ Large-range measuring system

Figure 3.7: The unfixed trackable pattern setup applying a highly accurate tracking system.

The fixed trackable pattern setup does not fully take the best advantage of the tracking system. Since the tracking system could track the position of the calibration boards, these two boards do not have to be rigidly linked anymore. This flexibility facilitates the improvement of measurement quality since each calibration pattern could be placed at positions relative to the corresponding cameras, which generate the best possible estimates. So an upgraded version of the fixed trackable pattern setup is presented, namely the unfixed trackable pattern setup. It is similar to the fixed trackable pattern setup, except that the two pattern boards are no longer linked and could be independently placed to different poses relative to cameras (Figure 3.7).

The measurement collection procedure is similar to the fixed trackable pattern setup. The two pattern boards *P1* and *P2* are independently placed to different poses relative to cameras, and a set of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ with better measurement quality together with $\{\mathbf{X}_i\}_{i=1}^{i=n}$ obtained from the tracking system is recorded. In the end, a final BA, including the weighting factors λ_i^{P1} and λ_i^{P2}, is applied to refine calibration results.

Since the relative pose $\mathbf{X}_i$ between the calibration patterns for each pose pair is accurately known from the tracking system, the replacement of $\mathbf{A}_i$ and $\mathbf{B}_i$ now becomes:

$$\begin{aligned}\mathbf{A}_i &= \mathbf{Y}\mathbf{B}_i\mathbf{X}_i^{-1}, \\ \mathbf{B}_i &= \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}_i.\end{aligned}$$

Since the above replacement of $\mathbf{A}_i$ and $\mathbf{B}_i$ still exists during the final refinement, the weighting factor could also be integrated in the same way as in Liu's setup and the fixed trackable pattern setup. However, it will not make much difference in the unfixed trackable pattern setup since the disconnection of the calibration patterns allows them to be captured with relatively good quality.

3.4 Validation on Simulated Dataset

In this section, the weighted non-linear optimization method and the data selection strategy are validated on the synthetic dataset. First, the explanation of how synthetic data is generated for Liu's setup is provided. The definition of error metrics that are going to be used for the evaluation of different algorithms is then presented. In the end, state-of-the-art methods with different settings are implemented and compared. The method of applying the highly accurate tracking system is not implemented in the simulation since there is no appropriate noise model for the tracking system.

3.4.1 Synthetic Dataset

As illustrated before, a customized calibration device is introduced to assist the calibration procedure, except that all the true transforms are precisely known in the simulation. Since no concrete research has been investigated on how to calculate the calibration pattern size and the relative pose $\mathbf{X}$ between two patterns, their values are determined through trial and error.

First, an exhaustive searching program is run based on all the known ground truth such as the relative pose between the calibration patterns $\mathbf{X}$, the relative pose between the camera pair $\mathbf{Y}$, the camera intrinsic parameters, etc. to produce a pose pair bank which consists of over 1,400 pose pairs. All pose pairs in the bank meet the following requirements. First, each pose pair in the bank is different from the rest both in translation and rotation so that the pose pairs in the bank discretely span the whole measurement space, which is continuous. Second, the projection size of the calibration pattern must exceed a certain threshold, which guarantees the minimum quality of the measurement. In this experiment, the threshold is set to 0.2 of the full image plane. The synthetic measurements are then generated based on the pose pair bank. The true pose pairs are first randomly extracted from the bank. The noise-free 2D coordinates obtained through the projection process are corrupted with Gaussian noise afterward. In the end, the noise-corrupted 2D coordinates are used as the measurement to recover the noisy pose pairs $\mathbf{A}_i$ and $\mathbf{B}_i$.

To demonstrate the influence of the spatial distribution of pose pairs on the calibration results, measurement sets with the following characteristics could be generated from the pose pair bank: (1) spatially scattered pose pair set with larger projection size; (2) spatially clustered pose pair set with larger projection size; (3) scattered distributed pose pair set with smaller projection size; (4) clustered distributed pose pair set with smaller projection size. The difference between the scattered distribution and the clustered one is that the former has both larger rotation and translation differences among all the pose pairs. Because all generated pose pairs are extracted from the bank, each pose pair in the clustered set has at least the same minimum translational and rotational difference as the ones in the bank. A similar criterion is used for measurement quality. The pose pairs that have larger projection sizes are extracted from the pose pair bank based on a more significant projection size threshold. Though the quality of the pose pair set with smaller projection size is bad compared to the pose pair set with large projection size, the former set is still guaranteed the minimum required quality since they are generated from the bank. The combination of the pose pair distribution and the projection size gives four extreme measurement sets, namely scattered large, clustered large, scattered small, and clustered small, where clustered and scattered suggest the pose distribution while large and small means the projection size of the resulting calibration pattern. The code for the calibration model, as well as the optimization strategies, is available online[2].

3.4.2 Error Metric

$\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ represent the estimated solutions which are calculated by applying different calibration methods. The ground truth of $\mathbf{X}$ and $\mathbf{Y}$ is known in the simulation environment, so the estimated $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ could be directly compared based on the following error metrics. Since the error metric calculation of $\mathbf{X}$ and $\mathbf{Y}$ is the same, only $\mathbf{X}$ is taken as an example.

Rotation Error

We apply the method of Wunsch et al. in [70] to define the rotation error. $\hat{\mathbf{q}}_X$ denotes the estimated quaternion of $\mathbf{X}$ and $\mathbf{q}_X$ the ground truth quaternion. The

[2] https://github.com/zaijuan/eye-to-eye-calibration.git

rotation error e_X^R is defined as:

$$e_X^R = min\{arccos(\mathbf{q}_X \cdot \hat{\mathbf{q}}_X), \pi - arccos(\mathbf{q}_X \cdot \hat{\mathbf{q}}_X)\}, \tag{3.16}$$

in which '$\cdot$' denotes the inner product of two quaternion vectors. Here the rotation error is represented by the angles returned by *arccos* and then mapped to $[0, 90°]$.

Translation Error

The estimated translation vector is described as $\hat{\mathbf{t}}_X$, and the ground truth is $\mathbf{t}_X$. The translation error is computed as follows,

$$e_X^t = \|\mathbf{t}_X - \hat{\mathbf{t}}_X\|. \tag{3.17}$$

3.4.3 Evaluation Results

In this part, the calibration results of different methods with different settings are presented. The calibration results of the method in [68], which is named as Wang's method after the author's family name, will be presented since all the other non-linear methods take its estimation of **X** and **Y** as the initial value.

To prove that applying the weighting factor during the optimization process alleviates the imbalance of the measurement quality, both the unweighted and weighted methods are implemented. The unweighted method does not utilize the weighting factor, while the weighted method applies the weighting factor. In parallel, the method of minimizing the reprojection error from only one calibration pattern is implemented for two reasons. The first reason is to test whether it gives better estimation results than Wang's method, which does not minimize the reprojection error. By comparing to Liu's method, the necessity of minimizing the reprojection error from both calibration patterns is validated.

The above methods are referred to as the unweighted two-side constrained method (Liu), the weighted two-side constrained method (Wgt-Liu), the unweighted one-side constrained method (Unwgt-1), and the weighted one-side constrained method (Wgt-1).

Besides different methods, it is also significant to show how their calibration robustness and accuracy correspond to the increase of image noise and various measurement numbers. In the first setting, the number of pose pairs changes from 5 to 45, with the fixed Gaussian noise of 1.0 pixels. In the second one, the added Gaussian noise on the image varies from 0.2 to 1.4 pixels, with a fixed number of 25 measurement pairs. The results shown below are taken the average of 100 iteration runs. For each iteration, the pose pairs are randomly extracted from the pose pair bank and processed applying different methods. Since the generation of each measurement set is random, 100 different measurement sets will be generated and used for the calibration procedure after repeating 100 times. Therefore, the demonstrated calibration results are the average of overall calibration error instead of a specific measurement set, which is objective and reliable.

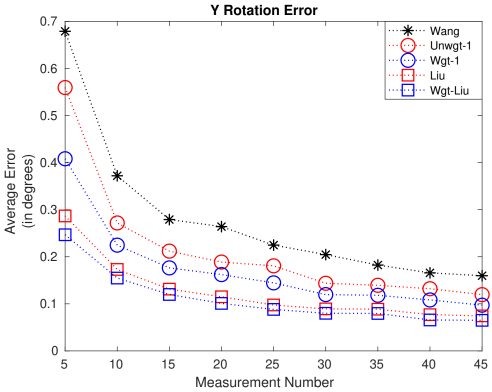


Figure 3.8: The rotation error of **Y** of different methods with increased pose pairs.

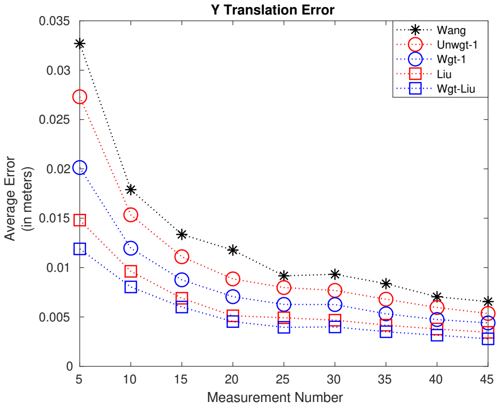


Figure 3.9: The translation error of **Y** of different methods with increased pose pairs.

The rotational and translational error of **Y** calculated from different methods with the increase of measurement number is demonstrated in Figure 3.8 and Figure 3.9. Since the calibration results of **X** are similar to **Y**, both in magnitude and pattern, it is unnecessary to present repetitive figures.

Several conclusions are drawn as follows. (1) The weighted two-side constrained method gives the best results. (2) In general, the methods either weighted or unweighted applying only one-side constraint result in larger errors than the methods applying two-side constraints, but smaller errors than the method without minimizing reprojection error. (3) The methods either one-side or two-side applying the weighting factor always give better results than the methods without the weighting strategy. (4) The calibration error of the methods minimizing the reprojection error keeps going down with the increase of the pose pair number.

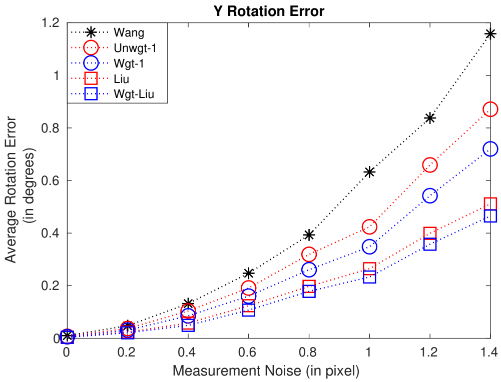


Figure 3.10: The rotation error of **Y** of different methods with increased image noise.

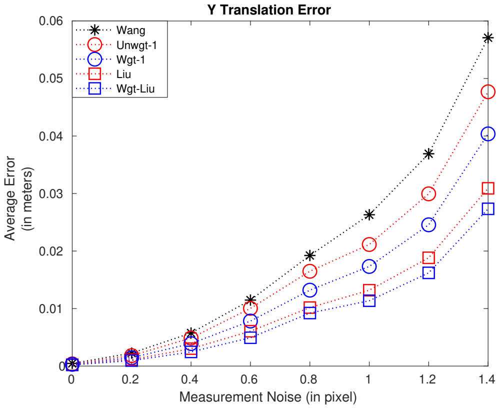


Figure 3.11: The translation error of **Y** of different methods with increased image noise.

Figure 3.10 and Figure 3.11 present calibration results of different methods with the increase of image noise, from which the following conclusions can be drawn. (1) The weighted two-side constrained method shows the least error under all noise conditions. (2) The performance of the weighted methods is better than the unweighted methods. (3) The two-side constrained methods produce smaller errors than the one-side constrained methods. (4) With the increase of measurement noise, the benefit of applying the weighting factor becomes more noticeable.

The above experiment results validate the improvement of the weighting factor in accuracy and robustness. With different measurement numbers, the Wgt-Liu method outperforms all other methods, and the calibration error keeps continuously going down with the increase of the measurement number. With the increase of image noise, the integration of the non-linear optimization method always generates the least calibration error.

When given a measurement set, the initial value influences the final estimation. It has also been validated in the simulation that the final estimation has been improved by choosing a subset, whose pose pairs are comparatively scattered, and whose measurements are of good quality. These results are not shown in the above figures because the improvement is not noticeable and would cover the results from Wgt-Liu. Instead, four extreme types of measurement sets, namely scattered good quality, clustered good quality, scattered bad quality, and clustered bad quality, are utilized to emphasize these differences depending on their spatial distribution and measurement quality. For each configuration, the measurement number is set to 25, and the noise level is 1.0 pixels.

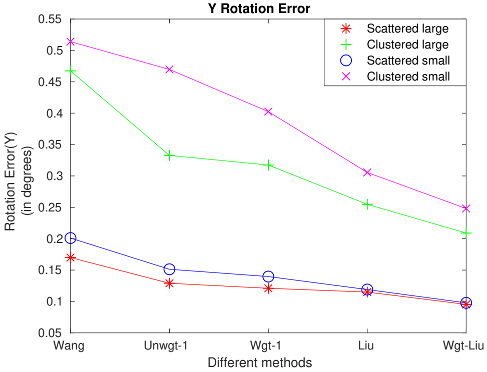


Figure 3.12: The rotation error of **Y** with regard to different pose pair configurations and different methods.

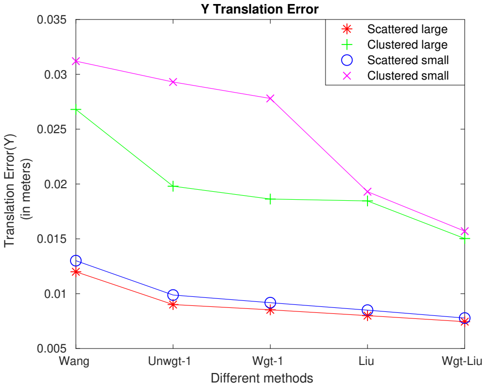


Figure 3.13: The translation error of **Y** with regard to different pose pair configurations and different methods.

The calibration results of these four different configurations of measurement sets that are applied with different calibration methods are demonstrated in Figure 3.12 and Figure 3.13. Same as the previous simulation, the calibration error is taken the average of 100 iteration runs, and the generated pose pair set is different for each run. Besides, all the extracted pose pair sets fulfill the corresponding characteristics of each configuration.

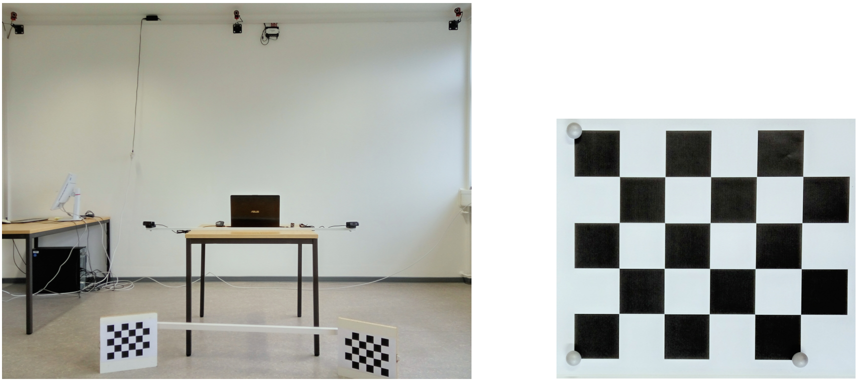
Two particular conclusions concerning these four different configurations are drawn as follows. First, for all methods, spatially scattered pose pair set with better measurement quality (larger projection size) generates the most accurate estimation, while spatially clustered pose pair set with smaller projection sizes leads to the worst estimation results. Second, spatially scattered pose pair set with smaller projection size produces better results than spatially clustered pose pair set with larger projection sizes regardless of different calibration methods. The first conclusion is evident. The second conclusion implies the calibration methods are more demanding on the distribution of pose pairs than their measurement quality.

The calibration results bring some insights into the tradeoff between the spatial distribution of pose pairs and their generated measurement quality. It is crystal clear that the combination of scattered pose pair distribution and larger projection size produces the best calibration results. However, these two factors are somehow mutually restricted. Scattered pose pair distribution implies the diversity of the projection size, while the demand for larger projection size limits the spatial distribution of pose pairs. This further explains why the introduced weighting factor is crucial during the optimization process. First, it increases the pose change space by allowing a larger varying range of the measurement quality. Second, the increased measurement space helps to provide a more accurate initial value from solving $\mathbf{AX} = \mathbf{YB}$, which is used for the following weighted non-linear refinement.

3.5 Real Experiment Results

3.5.1 Experiment Setup

In the real experiment, three different calibration setups presented in the previous sections are implemented: Liu's setup, the fixed trackable pattern setup, and the unfixed trackable pattern setup.



(a) The real experiment environment.

(b) The tracking targets.

Figure 3.14: The left figure demonstrates the real experiment environment. Above are equipped the high accuracy tracking system 'OptiTrack'. The camera pair with non-overlapping FOV is rigidly connected and fixed in the experiment, and the calibration device with two known planar patterns rigidly linked is placed on the ground. The right figure shows the calibration board used for detection by both the camera and the tracking system. The coordinate frame of the tracking targets and the pattern board coordinate system are aligned.

Figure 3.14a shows the real experiment environment, where a camera rig mounted with two cameras with non-overlapping FOV, an external calibration device, and an equipped highly accurate tracking system 'OptiTrack' are provided. The introduced calibration pattern boards used for recovering the relative pose to the camera pair could be accurately localized within the tracking system after aligning their coordinate frames with that of the tracking targets attached to them (Figure 3.14b).

The experiments are carried out as follows. As analyzed before, due to the underlying reduced and twisted properties of the pose change space in Liu's setup, the procedure of collecting a proper pose pair set is tricky. Instead of hesitating which pose pairs should be included, it is preferable to collect an abundant amount of pose pairs covering as much measurement space as possible. In the end, a set of images $\{\mathbf{I}_i^{P1}, \mathbf{I}_i^{P2}\}_{i=1}^{i=n}$ containing the planar calibration pattern used for recovering $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$, and the recovered $\mathbf{X}$ in the fixed trackable pattern setup or $\mathbf{X}_i$ in the unfixed trackable pattern setup are recorded. All the collected pose pairs are first filtered applying the data selection strategy, and a subset is then used to different optimization processes.

When a highly accurate tracking system is available in the calibration environment, the unfixed trackable pattern setup is preferred over the fixed trackable pattern setup since the former generates measurements with better quality than the latter one. By including the fixed trackable pattern setup in the real experiment, which is a mixed version of Liu's setup and the unfixed trackable pattern setup, the subtleties between different calibration setups and optimization methods could be better revealed.

3.5.2 Experimental Results

Unlike in the simulation, there is no ground truth in the real experiment. In order to evaluate the calibration results from different setups and verify the improvement brought by the weighted optimization method, the unfixed trackable pattern setup with the integration of the weighting optimization method serves as the benchmark since this configuration generates the best possible calibration results.

The same error criteria are used as in the simulation to evaluate the calibration difference of different methods. The benchmark is set as the weighted estimation of the unfixed trackable pattern configuration. The term difference is used in the real experiment instead of the previous term error to indicate that although the ground truth of $\mathbf{Y}$ is unknown, it could be estimated with the highest accuracy applying the unfixed trackable pattern configuration.

Figure 3.15 shows the calibration differences of different setups and different methods. Liu's method with the integrated optimization method generates a little larger but bearable calibration differences compared to the fixed trackable pattern configuration. Wang's method, which does not minimize the reprojection error, deviates the farthest from the benchmark. The performance of the fixed trackable

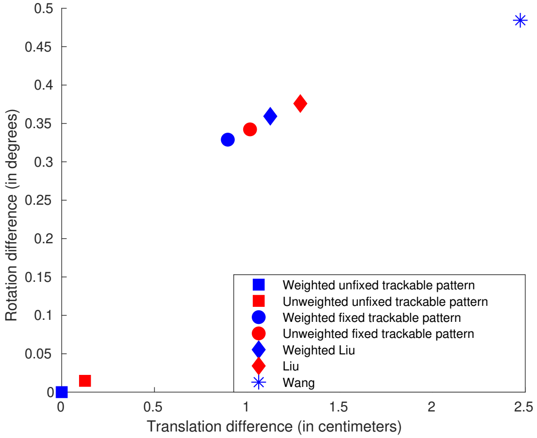


Figure 3.15: The calibration difference of **Y** concerning different setups and methods.

pattern setup lies between Liu's setup and the unfixed trackable pattern setup. In the unfixed trackable pattern setup, the difference between the weighted and the unweighted estimation is minor since all calibration patterns could be captured with relatively high resolution in this case. Nevertheless, applying the weighting factor generates less different results compared to the benchmark regardless of different setups and methods.

3.6 Conclusions and Discussion

In this chapter, a weighted non-linear optimization method, together with the data selection strategy, which is applicable to specific calibration setups are developed. The optimization method introduces the extra quality measure factor to the objective function, which increases the measurement space and improves the calibration accuracy. Hence, instability could be alleviated, and robustness could be safely guaranteed. Besides, by carefully choosing a measurement subset, the possibility of getting trapped in a worse local minimum is reduced.

During the simulation and the real experiment of Liu's setup, the appropriate size of the calibration patterns and the corresponding relative pose **X** between them are determined through trial and error. Since these two variables influence the measurement space and the measurement quality, they also affect the calibration results. In this thesis, how to determine the optimal values of these two variables is not investigated.

4 Eye-to-eye Calibration Applying Dynamic Fiducial Patterns

In this chapter, a new calibration method is proposed by introducing two electronic monitors for displaying dynamic fiducial patterns. The virtual pattern could actively regulate its configuration (size, position, structure) on the monitor during the calibration procedure so that measurements of better quality are generated and used for the calibration estimation. At first, the mechanism of the virtual pattern is explained, followed by the validation of the calibration method both in the simulation and in the real experiment.

4.1 Problem Statement

In the last chapter, the proposed weighted optimization method and the data selection strategy that could be integrated into specific setups such as Liu's setup as well as the unfixed trackable pattern setup have been investigated. Based on the data collection procedure and the optimization process, the error source of these two calibration setups could be summarized as follows: the estimation accuracy of $\mathbf{A}_i$ and $\mathbf{B}_i$, the pose change space, the method used for solving $\mathbf{AX} = \mathbf{YB}$, the method used for non-linear refinement. After the integration of the weighted optimization strategy to the specific calibration methods, both accuracy and robustness have been improved. This optimization method relieves the underlying reduced pose change space and the imbalance of the measurement quality, especially in Liu's setup. However, the optimization method does not create a larger pose change space; neither does it improve the measurement quality. The weighted optimization strategy passively puts weighting to all the measurements based on their quality. In consequence, the measurements with better quality will have a more significant effect on the estimation results. Considering the inherent limitations in Liu's setup, the calibration method applying the tracking system could accurately recover the relative pose between the calibration objects. Thus they could be disconnected and be independently placed to the camera rig. The weighted optimization method could also be applied to this setup, though the improvement is minor. From the perspective of the error source, the weighted optimization method improves the results by refining the final optimization process. In contrast, the unfixed trackable pattern setup reduces the first two error sources.

From the standpoint of accuracy, the method of applying a highly accurate tracking system is preferred if the laboratory is already equipped with such a tracking system. However, the cost of introducing a tracking system is high, and purchasing such a costly tracking system might not be worthy in some cases. Besides, the installation, calibration, and operation of the tracking system are complicated and time-consuming, which makes it a less preferable way to solve the eye-to-eye calibration problem merely. While from the perspective of calibration cost, Liu's method is preferred. However, Liu's method might not be accurate enough for some applications. This leads to the following inspiring question: since larger pose change space and good quality measurements bring better calibration estimation, is there an economical way to increase the measurement space as well as the measurement quality?

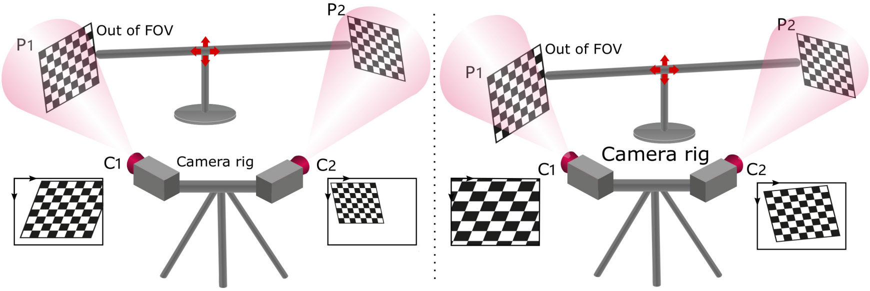


Figure 4.1: Two failure examples in Liu's setup. The captured calibration pattern *P1* in the left example lies a bit out of the image. In the right figure, because the calibration pattern *P1* is placed too close to the camera, only a part of it is captured even though the resolution, in this case, is very high (the whole image plane).

During the deduction of the optimization method in the last chapter, the twisted and compound effect of a minor pose change on the resulting images has been explained. In Figure 4.1, two failure examples are demonstrated. In both examples, the calibration patterns failed to be captured by the corresponding camera. Therefore, the relative pose between the camera and the calibration pattern could not be recovered, which makes these pose pairs invalid. However, once the relative pose could be recovered, the resulting measurement quality of these two pose pairs is quite good. This leads to the second question: considering the good measurement quality in these pose pairs, is there any way to recover the camera pose with regard to the calibration pattern?

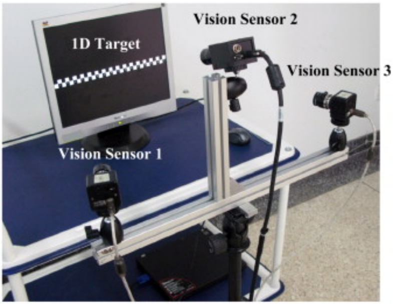


Figure 4.2: The method of solving the extrinsics of cameras with non-overlapping FOV using a 1D target imitated by a monitor [40].

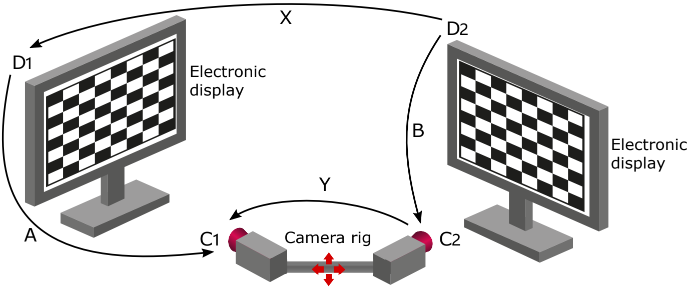


Figure 4.3: The proposed method applying dynamic patterns displayed on monitors.

The work in [40] solves the eye-to-eye problem using a 1D target imitated by a monitor (Figure 4.2). The rotation matrix between the camera pair is estimated by applying the co-linearity property of the feature points on the 1D target, and the translation vector is calculated based on the known distances between the feature points. In [2], the intrinsic parameters of the camera are calibrated using a curved display screen, from which dense feature points are generated and used for the intrinsic calibration. The advantages of introducing a display screen are multiple. First, the cost is low. Second, the number, the type, and the configuration of the fiducial features could be actively managed. Inspired by these advantages, a new calibration setup demonstrated in Figure 4.3 is proposed. The setup replaces the fix-sized calibration boards in Liu's setup with corresponding electronic displays. After proper encoding of the fiducial patterns displayed on the screen, the relative pose between the camera frame and the screen frame could still be recovered even if the camera captures only part of the screen. This advantage leads to a larger measurement space compared to Liu's setup.

Besides the increased measurement space, using monitors could also improve the measurement quality since the configuration of the fiducial patterns displayed on the monitor could be actively manipulated. The only remaining question is: how to actively generate these patterns yielding the most accurate estimation of $\mathbf{A}_i$ and $\mathbf{B}_i$?

4.2 Active Measurement

The measurement, based on which different estimations are calculated, is a set of sensor data representing the environment. Normally, interfering with the environment in order to obtain measurements that are more informative or less noise-corrupted is not possible. For example, LiDAR (Light Detection And Ranging) is widely mounted on cars for ADAS (Advanced Driver Assistance Systems). However, the performance is highly dependent on weather conditions. The emitting energy deteriorates under rainy or foggy conditions. Another example is the camera. The localization quality of detected features on an image is profoundly affected by the corresponding environmental feature points, which cannot be changed under most circumstances. This is why implanted fiducial landmarks are introduced in some applications: to provide measurements with more accuracy and robustness. The measurements in the above examples are either entirely passive or semi-active (in the case of fiducial landmarks) acquisitions from different sensors.

Active measurements, on the other hand, are obtained by actively creating a favorable environment, from which the sensor could better capture the features. The introduced electronic monitors in the new calibration setup can be used to actively display the fiducial landmarks, unlike the majority of the scenarios where the environment can not be modified. Compared to the semi-actively implanted fiducial landmarks, an active fiducial pattern could adapt its feature number, size, position, and the configuration in order for the camera to generate the best projection of the environment.

In the new eye-to-eye calibration setup (Figure 4.3), the aim of the active generation of predictable, reliable, and accurate fiducial features is to reach the best estimation of the camera pose, which serves the calibration method. Different P*n*P methods could be applied to estimate the camera pose, of which BA is the most accurate. For a well-calibrated camera, the objective function of estimating the camera pose $(\mathbf{R},\mathbf{t})$ using bundle adjustment is formulated as:

$$(\hat{\mathbf{R}},\hat{\mathbf{t}},\hat{\mathbf{P}}_1,\hat{\mathbf{P}}_2\cdots\hat{\mathbf{P}}_n) = \underset{\mathbf{R},\mathbf{t},\mathbf{P}_1,\mathbf{P}_2\cdots\mathbf{P}_n}{\arg\min}\sum_{i=1}^{i=n}\left\|\mathbf{p}_i-\frac{1}{\lambda_i}\mathbf{K}\left(\mathbf{R}\mathbf{P}_i+\mathbf{t}\right)\right\|_2^2.$$

Since the 3D position of the fiducial features is accurately known, it does not need to be optimized like in standard BA. Therefore, the above objective could be reduced to:

$$(\hat{\mathbf{R}},\hat{\mathbf{t}}) = \underset{\mathbf{R},\mathbf{t}}{\arg\min}\sum_{i=1}^{i=n}\|\boldsymbol{\varepsilon}_i\|_2^2 = \underset{\mathbf{R},\mathbf{t}}{\arg\min}\sum_{i=1}^{i=n}\left\|\mathbf{p}_i-\frac{1}{\lambda_i}\mathbf{K}\left(\mathbf{R}\mathbf{P}_i+\mathbf{t}\right)\right\|_2^2,$$

where $\boldsymbol{\varepsilon}_i$ represents the reprojection error of each 3D-2D correspondence. The objective function is a 2-norm, non-linear estimator, which comprises the sum of the reprojection errors resulted from all captured feature points. The to-be optimized camera pose is of six dimensions: three degrees of freedom for translation $\mathbf{t}$ and three degrees of freedom for rotation $\mathbf{R}$. Due to the special orthogonal property of the rotation matrix, the Lie algebra $\mathfrak{se}(3)$ is applied to describe the relative pose so as to transform the originally constrained optimization problem to an unconstrained one during the optimization process. Then it is feasible to apply non-linear optimization methods such as Gaussian-Newton [69], Levenber-Marquardt [46] etc. The camera pose $(\mathbf{R},\mathbf{t})$, when represented by the corresponding Lie algebra $\boldsymbol{\xi}$, stands for: $\mathfrak{se}(3) \doteq \left\{\boldsymbol{\xi} = \begin{bmatrix}\boldsymbol{\rho}\\ \boldsymbol{\phi}\end{bmatrix} \in \mathbb{R}^6, \boldsymbol{\rho}\in\mathbb{R}^3, \boldsymbol{\phi}\in\mathfrak{so}(3)\right\}$, where the translation and the rotation are depicted by $\boldsymbol{\rho}$ and $\boldsymbol{\phi}$, and $\mathfrak{so}(3)$ is the Lie algebra of Special Orthogonal Group SO(3). The above function could be transformed to the following applying Lie algebra:

$$(\hat{\boldsymbol{\xi}}) = \underset{\boldsymbol{\xi}}{\arg\min}\sum_{i=1}^{i=n}\|\boldsymbol{\varepsilon}_i\|_2^2 = \underset{\boldsymbol{\xi}}{\arg\min}\sum_{i=1}^{i=n}\left\|\mathbf{p}_i-\frac{1}{\lambda_i}\mathbf{K}\exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_i\right\|_2^2.$$

Based on this objective function, the estimation accuracy of the camera pose $\boldsymbol{\xi}$ depends on the following factors: the accuracy of 3D feature coordinates $\mathbf{P}_1,\mathbf{P}_2\cdots\mathbf{P}_n$, the number of the detected features n, the accuracy of the detected feature $\mathbf{p}_i$, the spatial configuration of 3D features.

Accuracy of the 3D feature coordinates One of the main advantages of introducing fiducial features is that their 3D coordinates are known a priori and are noise-free. In this case, the position of the 3D features does not need to be optimized, and the number of unknown variables is reduced. In contrast, the features reconstructed from the real environment usually contain unavoidable errors, even with high-quality sensors. Compared to building a high accuracy map of the environment using high-quality sensors, which is normally expensive, the inclusion of fiducial features is much cheaper.

Number of detected features n The number of the 3D-2D correspondences n will also influence the estimation. Generally, the more features are included, the more constraints they will bring into the cost function.

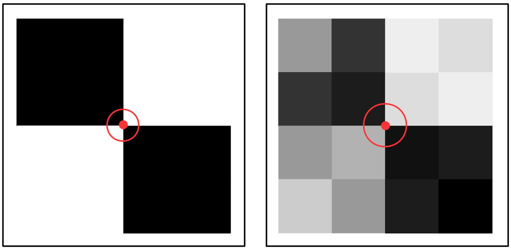


Figure 4.4: The left feature has rapid changes in the area, while the right feature has less image gradient. Thus, the left feature could be located with more accuracy than the right one regardless of which detector is applied.

Feature detection accuracy Many factors could influence the feature detection accuracy, such as camera quality, lighting in the environment, detection method, feature quality, etc. The features involved in this thesis are mainly interest points, which could be corners or blobs. Typical feature detectors are Harris [14], Shi-Tomasi [52], SUSAN [58], FAST [53], the Laplacian of Gaussian (LoG), the difference of Gaussians(DoG), etc. All these detectors aim at finding high curvature in the image gradient, except that the blob detectors could find features at an appropriate scale. Besides different detectors, the feature itself also makes a difference in the detection accuracy. Figure 4.4 provides an example of different features, of which the left feature has better quality since the sharper change in two directions allows detectors to localize the feature with more accuracy.

Spatial configuration of 3D features How 3D features are distributed in space also influences the estimation results. Figure 4.5 demonstrates two different configurations of four feature points, which will result in different pose estimation of the camera with respect to the world frame. At this point, the influence intro-

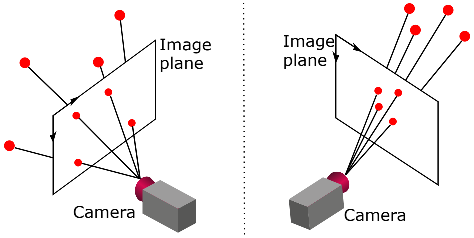


Figure 4.5: Two different configurations of four feature points. The configuration of the feature points in the left figure is scattered and decentralized. In contrast, the configuration in the right figure is clustered and centralized.

duced by the configuration is skipped. A thorough explanation will be given in the next section.

The remainder of this section focuses on the factors that are possible to apply to the new setup.

- **Feature quality** As analyzed before, interest points with high curvature result in the most accurate feature detection. Therefore, the fiducial patterns displayed on the screen will be an analog of the checkerboard. Though there are different types of artificial features, such as circles or ellipses that can be applied in this situation, the underlying generation, and configuration of these artificial features are similar. Besides, since all the calibration methods proposed in the last chapter use checkerboard patterns, the dynamic pattern will only concentrate on checkerboard-like patterns for consistency and better comparison.

- **Feature number** It is evident that the more feature points are generated, the more constraints they will bring to the cost function. From this standpoint, it is favorable to generate a fiducial pattern with dense feature points. However, the generation of dense feature points is not always possible. The new setup generates the fiducial pattern in an on-line manner, so the encoding of the 3D feature points displayed on the monitor is based on the estimated camera pose, which deviates from the ground truth. In this case, the robust encoding of dense feature points tolerates less noisy pose estimation, which is anti-causal since the intention of using fiducial patterns is to improve the estimation accuracy of the camera pose. An example is given in Figure 4.6. In the example, the feature points marked with red circles may not be captured by the camera, which will need an extra algorithmic processing block to create correct 3D-2D correspondences. So from the viewpoint of robustness and on-line performance, the fiducial features will not be densely distributed but constrained to the amount that guarantees the system's robustness.

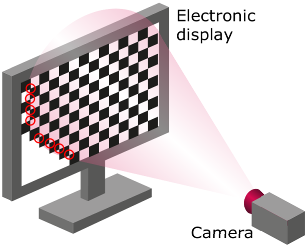


Figure 4.6: Demonstration of the difficulty in encoding dense features. The features within the circular cone are the ones that could be captured by the camera based on the estimated camera pose since the true camera pose is not known in the real experiment. Features that lie in the neighborhood of the cone border are marked with red circles. When the features are densely distributed, the presence of the noise in the camera pose estimation makes it challenging to encode those features because they might or might not be captured by the camera, which requires additional algorithmic tactics to ensure correct, robust 3D-2D correspondences.

- **Spatial configuration of fiducial features** How the fiducial pattern is distributed on the screen also has an effect on the pose estimation. The deduction of the dynamic pattern configuration will be explained in the next section.

4.3 Dynamic Pattern Configuration

As discussed in Section 4.1, one primary error source that exists in many eye-to-eye calibration methods is the estimation accuracy of the camera pose $(\mathbf{R}, \mathbf{t})$. In the last section, different factors that could influence the estimation accuracy are analyzed in detail. However, the assumption that different feature pattern configurations will lead to different camera pose estimates is proposed without concrete validation. Thus, a specific explanation will be presented in this section by analyzing the contribution of each error term in the objective function to the pose estimation.

Same as before, the pose $(\mathbf{R}, \mathbf{t})$ between the camera frame C with relative to the display screen D is estimated using BA. The pose is represented by the corresponding Lie vector $\boldsymbol{\xi}$, and the objective function is formulated as:

$$(\hat{\boldsymbol{\xi}}) = \arg\min_{\boldsymbol{\xi}} \sum_{j=1}^{m} \|\boldsymbol{\varepsilon}_j^D\|_2^2, \quad m \geq 3.$$

For clarity, the features in the above equation are numbered using j instead of commonly used i in order to be consistent with the definition of i, which is referred to as the pose pair number within the calibration context. Since $\boldsymbol{\varepsilon}_j^D$ is 2-dimensional and $\boldsymbol{\xi}$ is a 6×1 vector, the corresponding Jacobian is a 2×6 matrix. For simplicity, only one general term $\boldsymbol{\varepsilon}_j^D$ from the overall sum is considered since all error terms share the same structured Jacobian matrix. Based on the chain rule, the derivative of the term $\boldsymbol{\varepsilon}_j^D$ with respect to the relative pose $\boldsymbol{\xi}$ is described as follows:

$$\boldsymbol{J}_{\boldsymbol{\xi}}^{\boldsymbol{\varepsilon}_j^D} = \frac{\partial_{\boldsymbol{\varepsilon}_j^D}}{\partial \boldsymbol{\xi}} = \frac{\partial \boldsymbol{\varepsilon}_j^D}{\partial \mathbf{P}_j^C} \frac{\partial \mathbf{P}_j^C}{\partial \boldsymbol{\xi}}.$$

The first term is the derivative of the reprojection error $\boldsymbol{\varepsilon}_j^D$ with respect to the 3D feature points $\mathbf{P}_j^C$ represented in the camera frame C, and the second term represents the derivative of the transformed feature points $\mathbf{P}_j^C$ with regard to the change of the relative pose $\boldsymbol{\xi}$. These two components will be separately derived and concatenated afterward.

Applying Lie algebra, the feature point $\mathbf{P}_j^C$ in the camera frame C is related to the same feature point $\mathbf{P}_j^D$ in the display frame D by the equation:

$$\mathbf{P}_j^C = (\exp(\boldsymbol{\xi}^{\wedge})\mathbf{P}_j^D)_{1:3} = \begin{bmatrix} X_j^C & Y_j^C & Z_j^C \end{bmatrix}^T. \tag{4.1}$$

After applying the pinhole camera model, the following equation could be obtained:

$$Z_j^C \begin{bmatrix} x_j^D \\ y_j^D \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{P}_j^C = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_j^C \\ Y_j^C \\ Z_j^C \end{bmatrix}. \tag{4.2}$$

The $\mathbf{K}$ appearing in the equation is the camera matrix, which consists of the intrinsic parameters: f_x, f_y are the size of unit length in horizontal and vertical pixels, and c_x, c_y are the coordinates of the principal point in pixels along x and y axis. All the values of the above variables could be accurately obtained after the intrinsic camera calibration.

Based on the above equations (4.1) (4.2), the derivative of the first term could be formulated as:

$$\frac{\partial \boldsymbol{\varepsilon}_j^D}{\partial \mathbf{P}_j^C} = -\begin{bmatrix} \frac{\partial x_j^D}{\partial X_j^C} & \frac{\partial x_j^D}{\partial Y_j^C} & \frac{\partial x_j^D}{\partial Z_j^C} \\ \frac{\partial y_j^D}{\partial X_j^C} & \frac{\partial y_j^D}{\partial Y_j^C} & \frac{\partial y_j^D}{\partial Z_j^C} \end{bmatrix} = -\begin{bmatrix} \frac{f_x}{Z_j^C} & 0 & -\frac{f_x X_j^C}{Z_j^{C^2}} \\ 0 & \frac{f_y}{Z_j^C} & -\frac{f_y Y_j^C}{Z_j^{C^2}} \end{bmatrix}. \tag{4.3}$$

The derivative of the second term is deduced as follows. A direct derivation could bring in an additional term, which is used to relate the transform between Lie algebra and Lie group. Besides, the perturbation model is used instead to keep the derived Jacobian matrix clean and compact. The perturbation model first adds a tiny perturbation onto the Lie group, and the derivative is conducted concerning its corresponding Lie algebra [59]. Therefore, the derivative of the second term could be transformed into the following:

$$\frac{\partial \mathbf{P}_j^C}{\partial \boldsymbol{\xi}} \doteq \frac{\partial \mathbf{P}_j^C}{\partial \delta\boldsymbol{\xi}} = \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\mathbf{P}_j^C(\delta\boldsymbol{\xi} \bigoplus \boldsymbol{\xi})}{\delta\boldsymbol{\xi}}.$$

The '$\bigoplus$' in the equation is the 'addition' of the tiny perturbation onto the left side of $\boldsymbol{\xi}$. The derivation could then be deduced as follows. $\mathbf{P}_j^C$ and $\mathbf{P}_j^D$ in the equations are of their homogeneous forms:

$$\begin{aligned} \frac{\partial \mathbf{P}_j^C}{\partial \boldsymbol{\xi}} &= \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\exp(\delta\boldsymbol{\xi}^\wedge)\exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_j^D - \exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_j^D}{\delta\boldsymbol{\xi}} \\ &\approx \lim_{\delta\boldsymbol{\xi}\to 0} \frac{(\mathbf{I} + \delta\boldsymbol{\xi}^\wedge)\exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_j^D - \exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_j^D}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\delta\boldsymbol{\xi}^\wedge \exp(\boldsymbol{\xi}^\wedge)\mathbf{P}_j^D}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\begin{bmatrix} \delta\boldsymbol{\phi}^\wedge & \delta\boldsymbol{\rho} \\ \mathbf{0}^\mathrm{T} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}\mathbf{P}_j^D + \mathbf{t} \\ 1 \end{bmatrix}}{\delta\boldsymbol{\xi}} \\ &= \lim_{\delta\boldsymbol{\xi}\to 0} \frac{\begin{bmatrix} \delta\boldsymbol{\phi}^\wedge(\mathbf{R}\mathbf{P}_j^D + \mathbf{t}) + \delta\boldsymbol{\rho} \\ 0 \end{bmatrix}}{\delta\boldsymbol{\xi}} \\ &= \begin{bmatrix} \mathbf{I} & -(\mathbf{R}\mathbf{P}_j^D + \mathbf{t})^\wedge \\ \mathbf{0}^\mathrm{T} & \mathbf{0}^\mathrm{T} \end{bmatrix}. \end{aligned}$$

Same as before, the '$\wedge$' operator transforms a vector to its corresponding skew-symmetric matrix. The first three columns correspond to the translational derivative, and the rest three columns correspond to the rotational part. Since the last entry of the homogeneous coordinate always equals 1, the above equation could be reduced to:

$$\frac{\partial \mathbf{P}_j^C}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \mathbf{I} & -\mathbf{P}_j^{C\wedge} \end{bmatrix}. \tag{4.4}$$

Concatenating these two derivative terms from Eq. (4.3) and Eq. (4.4), the final Jacobian matrix is constructed as:

$$\boldsymbol{J}_{\boldsymbol{\xi}}^{\boldsymbol{\varepsilon}_j^D} = -\begin{bmatrix} \frac{f_x}{Z_j^C} & 0 & -\frac{f_x X_j^C}{Z_j^{C2}} & -\frac{f_x X_j^C Y_j^C}{Z_j^{C2}} & f_x + \frac{f_x X_j^{C2}}{Z_j^{C2}} & -\frac{f_x Y_j^C}{Z_j^C} \\ 0 & \frac{f_y}{Z_j^C} & -\frac{f_y Y_j^C}{Z_j^{C2}} & -f_x - \frac{f_y Y_j^{C2}}{Z_j^{C2}} & \frac{f_y X_j^C Y_j^C}{Z_j^{C2}} & \frac{f_y X_j^C}{Z_j^C} \end{bmatrix}.$$

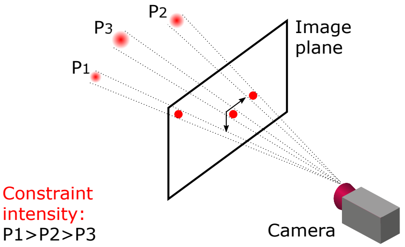


Figure 4.7: Demonstration of the sensitivity of the spatial configuration of the feature points to the reprojection error. In order to reduce the same amount of the reprojection error, the feature points with larger X^C, Y^C ,and smaller Z^C bring stronger constraints. In this example, P_1, P_2, and P_3 have similar depth coordinates Z^C. However, P_3 has the smallest X^C and Y^C, while P_1 has the largest X^C and Y^C. Therefore, the intensity level of the contributed constraints ranging from high to low is: P_1, P_2, P_3.

The absolute value of the entries in the Jacobian matrix indicates the sensitivity to the change of the objective function with respect to its variables, in this case, the reprojection error $\boldsymbol{\varepsilon}_j^D$ with regard to the relative pose $\boldsymbol{\xi}$. The following consistent conclusion could then be drawn based on the analysis of all entries in Jacobian matrix $\boldsymbol{J}_{\boldsymbol{\xi}}^{\boldsymbol{\varepsilon}_j^D}$. First, since f_x and f_y are the intrinsic parameters of the camera, their influence is determinate and could not be changed. Second, the feature points with comparatively larger X_j^C, Y_j^C, and smaller Z_j^C produce larger gradient values, which indicates those points are more sensitive to the pose change. The visualization of this conclusion is shown in Figure 4.7.

In this new calibration setup where two electronic displays are introduced, the depth of the fiducial features from each image only varies within a very limited range, so the variations in X and Y direction will have the dominating influence on the estimation results. When feature points with larger X_j^C, Y_j^C and similar Z_j^C are projected onto the image plane, their corresponding 2D points are the ones lying on the outer area of the image plane. Another way to interpret this new calibration setting is: if two images are detected with the same amount of features that have similar depth, the image whose feature points are more decentralized spread provides stronger constraints when compared to the image whose features are comparatively clustered towards its image center. Hence, the former is preferred for more accurate pose estimation.

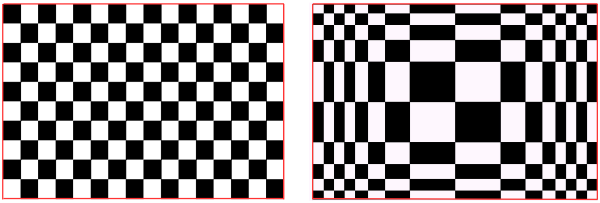


Figure 4.8: Two different image feature configurations used for the generation of the dynamic fiducial pattern through back-projection. The amount of the features on both images is the same, except that the features on the left side image are evenly spread, while the features on the right side are decentralized distributed.

Despite the improper use of ‘decentralized’, in what follows, two different image distributions shown in Figure 4.8 are referred to as decentralized and centralized to be indicative of the characteristics of their image pattern distributions. Since the features of the decentralized pattern are unevenly distributed on the image, the uneven pattern and the decentralized pattern are used interchangeably to refer to the same feature pattern. Similarly, the evenly distributed pattern and the centralized distributed pattern are the same.

The principle of the dynamic pattern configuration is based on the above deduction. First, the prospective configuration of the 2D image features that are decentralized distributed on the image plane is built, which provides the maximum constraints for pose estimation as analyzed before. These 2D feature points are then back-projected onto the monitor plane, and their corresponding 3D coordinates could be acquired. In the end, the dynamic fiducial pattern is constructed to create the expected 3D feature structure. In other words, actively regulating the configuration of the fiducial pattern on the monitor is accomplished through the back-projection of the requested 2D image feature distribution. In situations where the fiducial pattern on display could not fill up the whole image plane, only the part of the pattern that appears within the FOV of the corresponding camera is applied for pose estimation.

4.4 Calibration Setups Applying Dynamic Fiducial Patterns

In this section, different calibration setups applying dynamic fiducial patterns are presented based on the mechanism explained in the last section to reach optimal dynamic pattern configurations. The fiducial pattern could actively adjust its size, position as well as the structure on the electronic display based on the estimated relative pose between the camera and the monitor. Depending on whether the camera rig is movable or not, two calibration setups are proposed.

4.4.1 Calibration Setup for Movable Camera Rig

When the camera rig is movable, the following setups could be implemented, applying dynamic fiducial patterns.

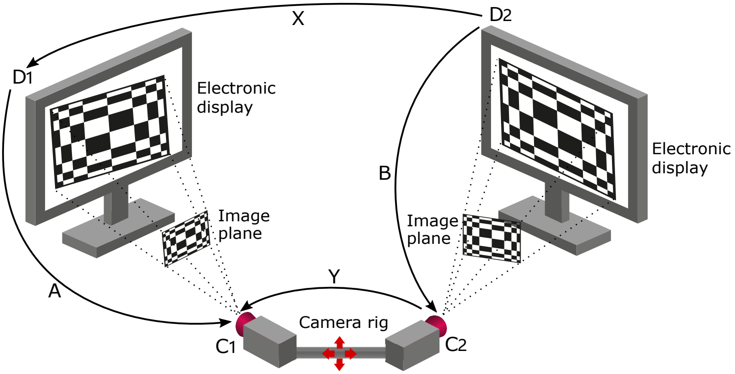


Figure 4.9: The calibration setup applying dynamic fiducial patterns. Two electronic displays are introduced for demonstrating dynamic fiducial patterns. The camera rig needs to be placed at several different poses (at least four) with regard to the monitors. During this procedure, the virtual pattern could dynamically regulate its configuration (size, position, and structure) on the monitor based on the estimated pose between the monitors and the corresponding cameras, which provides better-quality measurements and larger pose change space compared to using fixed-sized calibration patterns like in [39].

The proposed calibration setup necessitates two electronic displays to demonstrate the fiducial patterns (Figure 4.9). The camera rig is placed at different poses with regard to the monitors. For each pose pair, the fiducial pattern actively changes its configuration on the monitor based on the initial estimation of the camera pose. Since the calibration procedure is performed in real time, the estimation of the camera pose from the last time step is taken as the initial estimation of the current time step. As long as the camera is moving at a smooth and slow motion, the fiducial pattern will be generated with unnoticeable deviations from the prospective fiducial pattern. This deviation will not introduce any error. It only indicates that there is a minor difference between the optimal image feature distribution and the practically obtained one.

In the end, a set of images $\{\mathbf{I}_i^{D1}, \mathbf{I}_i^{D2}\}_{i=1}^{i=n}$ containing dynamic patterns and the corresponding fiducial pattern information are collected for estimating the unknown extrinsic parameters $\mathbf{X}$ and $\mathbf{Y}$, where $\mathbf{X}$ is the relative pose between the introduced displays and $\mathbf{Y}$ depicts the relative pose between the camera rig. The initial estimation of $\mathbf{X}$ and $\mathbf{Y}$ is provided by solving $\mathbf{AX} = \mathbf{YB}$, which is used to minimize the reprojection error based objective function. The deduction of the optimization process is not repeated since it is similar to Liu's method. The final optimization problem is formulated as follows after combining all the constrained projections:

$$(\hat{\mathbf{R}}_X, \hat{\mathbf{t}}_X, \hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_X, \mathbf{t}_X, \mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\sum_{j=1}^{m(i)} \|\boldsymbol{\varepsilon}_{ij}^{D1}\|_2^2 + \sum_{l=1}^{o(i)} \|\boldsymbol{\varepsilon}_{il}^{D2}\|_2^2).$$

Compared to Liu's method, this new setup has the following differences. Firstly, the 3D position of the fiducial features and the feature numbers $m(i)$, $o(i)$ vary from pose to pose. Secondly, the increased pose change space will lead to a more accurate initial estimation of $\mathbf{X}$ and $\mathbf{Y}$ from solving $\mathbf{AX} = \mathbf{YB}$. Lastly, the camera pose $\mathbf{A}_i$ and $\mathbf{B}_i$ are still constrained by the 3D closed-loop transformation equation, so they will be replaced with $\mathbf{X}$ and $\mathbf{Y}$ for the non-linear refinement. Since the estimation accuracy of $\mathbf{A}_i$ and $\mathbf{B}_i$ directly influences the following estimation accuracy of $\mathbf{X}$ and $\mathbf{Y}$ after this replacement, applying the dynamic pattern, in this case, reduces the error propagation by providing better estimated $\mathbf{A}_i$ and $\mathbf{B}_i$.

Due to the replacement of $\mathbf{A}_i$ and $\mathbf{B}_i$ for the final refinement, the estimation accuracy of $\mathbf{A}_i$ and $\mathbf{B}_i$ directly influences the estimation accuracy of $\mathbf{X}$ and $\mathbf{Y}$. To reduce the error propagation introduced by the above replacement, the weighted optimization strategy is integrated to 4.4.1, which leads to the following objective:

$$(\hat{\mathbf{R}}_X, \hat{\mathbf{t}}_X, \hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_X, \mathbf{t}_X, \mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\lambda_i^{D1} \sum_{j=1}^{m(i)} \|\boldsymbol{\varepsilon}_{ij}^{D1}\|_2^2 + \lambda_i^{D2} \sum_{l=1}^{o(i)} \|\boldsymbol{\varepsilon}_{il}^{D2}\|_2^2),$$

where λ_i^{D1} and λ_i^{D2} are the weighting factors used for the reprojection error related to the dynamic pattern *D1* and *D2* respectively. Their values are calculated as follows:

$$\begin{aligned} \lambda_i^{D1} &= \sqrt{S_i^{D2}/S_{max}}, \\ \lambda_i^{D2} &= \sqrt{S_i^{D1}/S_{max}}, \end{aligned}$$

in which S_i^{D1} and S_i^{D2} are the projection size of the corresponding dynamic pattern *D1*, *D2*, and S_{max} represents the full image size.

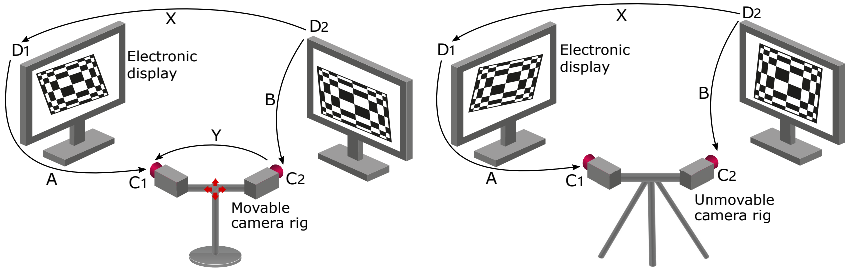
When it is possible to fix the camera rig during the calibration procedure, the estimation accuracy may be improved further. In this case, the camera pose estimated from the current fiducial pattern could be applied to re-generate a new fiducial pattern, which is used in turn to update the estimation of the camera pose. This estimation circle terminates if the difference between the two consecutive camera pose estimations is less than the predefined threshold.

Compared to Liu's setup, these calibration setups generate more accurate results by providing larger movement space and measurement with better quality. However, since the screen could not be moved during the calibration process, the methods could not be *directly* applied to situations where it is not convenient or possible to move the camera rig.

4.4.2 Calibration Setup for Unmovable Camera Rig

When the camera rig cannot be moved during the calibration procedure, either additional equipment is introduced in order to fix and assist the movement of the monitors, or a pre-calibration to estimate the relative pose **X** between the display screens needs to be implemented first.

In the first case, the calibration procedure is similar to the setup for mobile camera rig except that the monitors are movable in this situation. In the second case, the pre-calibration procedure needs an external assisting device: an external movable camera rig (Figure 4.10a). The data collection procedure, as well as the optimization method, is the same as the method proposed for the movable camera rig. After the pre-calibration, the relative pose **X** between the displays could be accurately recovered. Then the to-be-calibrated camera rig is placed in



(a) Pre-calibration.

(b) After the pre-calibration.

Figure 4.10: Eye-to-eye calibration method applying dynamical fiducial patterns for the camera rig, which is not movable during the calibration. The left figure demonstrates the pre-calibration procedure in which an external device is introduced to recover the relative pose **X** between the screens. After the accurate estimation of the relative pose **X**, the camera rig is placed in an appropriate position to the screens, and the relative pose **A** and **B** could be obtained. The unknown relative pose **Y** could thus be solved.

front of the electronic displays, and the relative pose $\mathbf{A}_i$ and $\mathbf{B}_i$ could be obtained (Figure 4.10b). Since **X** is known, the initial value of **Y** could be calculated based on the closed-loop $\mathbf{AX} = \mathbf{YB}$ and further refined based on the following objective function:

$$(\hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\lambda_i^{D1} \sum_{j=1}^{m(i)} \|\boldsymbol{\varepsilon}_{ij}^{D1}\|_2^2 + \lambda_i^{D2} \sum_{l=1}^{o(i)} \|\boldsymbol{\varepsilon}_{il}^{D2}\|_2^2),$$

where **X** is known from the pre-calibration and does not need to be optimized. Same as before, λ_i^{D1} and λ_i^{D2} are the weighting factors. During the calibration procedure, the fiducial patterns are actively generated in order to reach the most accurate estimation.

4.4.3 Comparison to Weighted Liu's Method

Since the calibration setups applying dynamic fiducial patterns are upgraded from Liu's setup, it is necessary to compare these two setups.

Both Liu's method and the methods applying dynamic fiducial patterns are calibration-friendly in terms of cost, simplicity, and convenience. When the laboratory is not equipped with a highly accurate tracking system and the cameras are not going to be applied to extremely high-demanding tasks, both Liu's method and the methods of using dynamic fiducial patterns could be implemented. Fiducial features either introduced statically in Liu's setup or dynamically in the dynamic fiducial pattern setups are error-free and could be further included for final refinement.

Compared to Liu's method that uses fixed-sized planar calibration patterns, the methods applying dynamic fiducial patterns could create tremendously larger measurement space and better measurement quality by dynamically adjusting the size and position of the fiducial patterns based on the estimated pose between the camera and the corresponding monitor, which in turn improves calibration accuracy.

In the last chapter, the inconvenience caused by determining the suitable calibration pattern size and the relative pose $\mathbf{X}$ between calibration patterns was discussed. A short summary is as follows. First, these two variables are dependent. Second, since they influence the measurement space and measurement quality, they will also affect the calibration result; Lastly, the values are determined from trial and error since no solid theoretical study has been investigated. In contrast, the setup of using electronic displays, in this case, does not have to determine the size of the pattern since the size of the pattern is dynamically changing depending on the relative pose $\mathbf{A}_i$ and $\mathbf{B}_i$. In order to achieve the optimal performance, the calibration device in Liu's setup has to be re-configured and re-manufactured to solve different eye-to-eye configurations. In contrast, the re-configuration of the displays could be easily accomplished by adjusting their relative pose.

From the perspective of optimization, the dynamic fiducial pattern is generated in order to provide maximum constraints for camera pose estimation. Hence, more accurate $\mathbf{A}_i$, $\mathbf{B}_i$ are estimated in the dynamic fiducial pattern setup compared to Liu's setup. Better estimated $\mathbf{A}_i$ and $\mathbf{B}_i$ would reduce error propagation into the non-linear optimization after the replacement $\mathbf{A}_i = \mathbf{Y}\mathbf{B}_i\mathbf{X}^{-1}$ and $\mathbf{B}_i = \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}$. The dynamic pattern here serves as a local structural weighting factor, which automatically influences the underlying optimization process. In contrast, Liu's method uses the numerical weighting factors based on the measurement quality to limit the error proposed passively. When it comes to the global objective function, the dynamic pattern is functioning as a global structural weighting factor. The varying importance of all fiducial feature points is delivered based on the contribution or the constraints they provide to the objective function. The stronger constraints they bring, the more they will be emphasized. In other words, the contribution of each fiducial feature is '**normalized**' from a global point of view, depending on their inherent structure. Therefore, no specific numerical weighting factors like in Liu's method are incorporated into each error item, which facilitates the optimization procedure.

However, the method applying dynamic fiducial patterns also has its limitations. The method is sensitive to the lighting due to the presence of the display screens, so the calibration is constrained to the indoor environment, where the lighting could be easily controlled. For the calibration situation where the camera rig is not able to move, additional equipment is necessary to fix and assist the movement of the electronic screens. To get around the limitation where the electronic screens and the camera rig are inconvenient to move during the calibration process, the relative pose between the screens could be first recovered through pre-calibration by introducing an external camera rig. Nevertheless, this extra camera rig decreases the flexibility and increases the calibration complexity.

4.5 Evaluation on Synthetic Dataset

In this section, different calibration setups and configurations are implemented in the simulation. They are then compared and analyzed.

4.5.1 Experiment Setup and Configuration

The calibration method applying dynamic fiducial patterns is compared with Wang's method, Liu's method with the integration of weighting factors (weighted Liu's method). By comparing with the weighted Liu's method, the intention is to verify the improvement brought by including the dynamic patterns during the calibration procedure. Meanwhile, in order to prove the dynamic virtual pattern constructed by back-projecting the deduced decentralized image feature distribution generates more accurate pose estimation, the centralized image feature distribution is tested and compared. Besides, the dynamic fiducial pattern method with the weighted optimization strategy integrated is implemented similar to the method without the weighting strategy with the following two intentions: to prove the necessity of the weighted optimization strategy and to compare with the improvement brought by the other factor, namely different pattern configurations. Since Wang's method could be applied to all the above setups and configurations and provides an initial value of $\mathbf{X}$ and $\mathbf{Y}$ for the non-linear refinement, only the result estimated based on the unevenly distributed dynamic fiducial pattern setup is demonstrated. By comparing Wang's result with the weighted Liu's method, the interesting comparison between the improvement brought by the dynamic fiducial pattern without non-linear refinement and the improvement brought by the final refinement based on the fixed-sized calibration pattern could be revealed. These setups and configurations are referred to as Wang, Wgt-Liu, Unwgt-Even, Wgt-Even, Unwgt-Uneven, Wgt-Uneven.

In order to impartially compare the calibration accuracy of different calibration setups and configurations, the same camera rig is used. In the dynamic fiducial pattern setup, two 24-inch electronic monitors with a resolution of 1900×1200 are introduced for displaying the dynamic fiducial patterns. At the same time, two fixed-sized checkerboards whose size is the same as the electronic displays are used for Liu's setup.

All calibration methods necessitate a set of different pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ for estimating the unknowns. Two different pose banks, one for Liu's setup, and the other for the dynamic fiducial pattern setup, are generated based on all the ground truth and camera parameters. The generation of the pose pair bank for the dynamic fiducial pattern setup is similar to Liu's setup, except that the fullscreen does not have to be captured in the former setup. The calibration configuration for Liu's setup and the dynamic fiducial pattern setup are slightly different. The difference lies in the ground truth $\mathbf{X}^{true}$, which is carefully re-adjusted for Liu's method considering the limited pose change space caused by the fixed-sized calibration patterns. All the pose pairs in the pose bank are different from each other, and they all generate images on which the projection size of the fiducial pattern exceeds 0.2 of the full image. The former reduces the potential instability, and the latter guarantees the minimum required measurement quality.

Based on the noise-free pose pair bank, the synthetic measurements for different setups are generated as follows. At first, a set of true pose pairs $\{\mathbf{A}_i^{true}, \mathbf{B}_i^{true}\}_{i=1}^{i=n}$ is randomly extracted from the corresponding bank. In Liu's case, the 3D coordinates of all fiducial features are fixed and known, so the noise-free 2D projections of the corresponding fiducial features could be directly obtained. Afterward, Gaussian image noise is added to get noise-corrupted 2D coordinates, which are used to produce the estimated pose pairs $\{\mathbf{A}_i^{est}, \mathbf{B}_i^{est}\}_{i=1}^{i=n}$. In contrast, in the dynamic fiducial pattern setup, $\mathbf{A}_i^{true}$ and $\mathbf{B}_i^{true}$ are first applied to find the virtual pattern on the monitor based on the principle described in the last section, which varies depending on the specific pose between the camera and the monitor. The obtained 2D error-free image coordinates are corrupted with Gaussian image noise and used for calculating the noisy pose pairs $\{\mathbf{A}_i^{est1}, \mathbf{B}_i^{est1}\}_{i=1}^{i=n}$, which are then applied to re-generate the virtual pattern. The projections of those fiducial patterns computed from the estimated pose pairs ($\mathbf{A}_i^{est1}$, $\mathbf{B}_i^{est1}$) are again contaminated by Gaussian image noise and used for estimating $\{\mathbf{A}_i^{est2}, \mathbf{B}_i^{est2}\}_{i=1}^{i=n}$, which are taken as the final estimated pose pairs for the dynamic fiducial pattern setup. In other words, the generated dynamic fiducial pattern is based on the estimated pose pairs rather than their ground truth, which ensures that the following calibration results concerning different setups are similar to real-world calibration conditions and convincing.

4.5.2 Evaluation of Different Methods and Configurations

In this part, the comparison of single pose estimation accuracy after applying decentralized and centralized image patterns is first presented, followed by the comparison of different setups and configurations with regard to the increased number of pose pairs and different image noise levels.

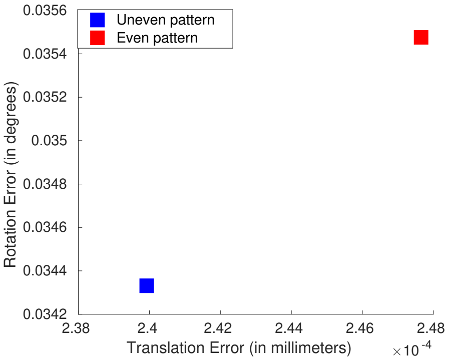


Figure 4.11: The average rotation and translation error of two different pattern types: evenly distributed feature pattern, and unevenly, decentralized distributed feature pattern.

Before demonstrating the benefits of applying the dynamic fiducial pattern to eye-to-eye calibration, it is essential and necessary to validate the improvement on single pose estimation. Two different image feature distributions, namely centralized and decentralized, are applied to 500 different poses, which are randomly extracted from the bank and processed with 1.0-pixel image noise. Figure 4.11 shows the average rotation error and translation error with respect to different pattern configurations, from which the pose estimation applying unevenly distributed pattern generates less error.

The second experiment is implemented as follows. Different sets of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ with the measurement number changing from 5 to 45 are first randomly extracted from the corresponding pose pair bank and corrupted with fixed Gaussian image noise $\boldsymbol{\varepsilon} = 1.0$ pixels. Wang's method, the weighted Liu's method, and the dynamic fiducial pattern methods with different pattern configurations are applied.

The calibration results shown in Figure 4.12 and Figure 4.13 are taken an average of 100 runs.

The following conclusions could be drawn. (1) The weighted decentralized dynamic pattern method shows the best results. (2) The methods applying weighted optimization always generate better results than unweighted methods. (3) The decentralized dynamic pattern configuration outperforms the centralized dynamic pattern configuration since the former produces less deviation from the ground truth. (4) Wgt-Even method results in less error than Unwgt-Uneven,

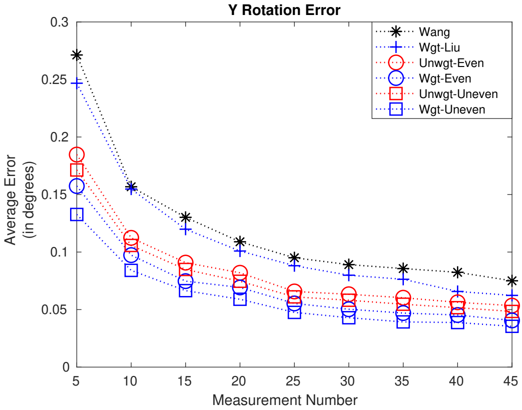


Figure 4.12: The rotation error of **Y** of different methods and configurations with increased pose pairs.

which indicates that the weighted optimization strategy has a larger contribution to the improvement of the calibration results compared to the configuration of the dynamic fiducial pattern. (5) All the methods applying dynamic fiducial patterns generate better results than the weighted Liu's method, even the translation error from Wang's method, which does not minimize the reprojection error, is less than the weighted Liu's method.

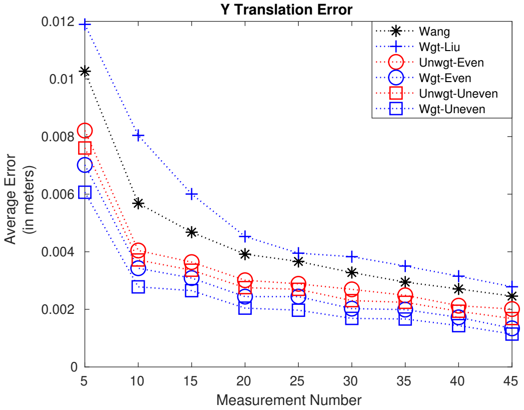


Figure 4.13: The translation error of **Y** of different methods and configurations with increased pose pairs.

All the setups and configurations in the third experiment are the same as the last one except that the measurement number is set to be 25, while the Gaussian image noise varies from 0.2 to 1.4 pixels. Same as before, the results are taken an average of 100 iteration runs.

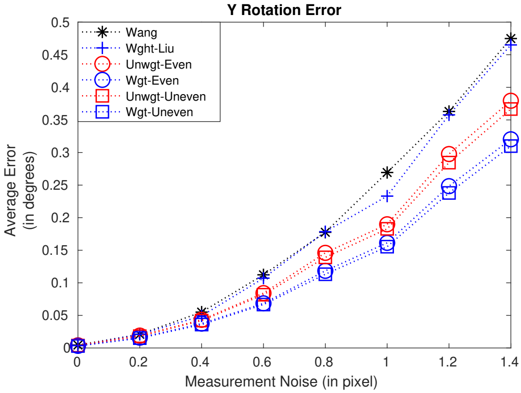


Figure 4.14: The rotation error of **Y** of different methods and configurations with increased image noise.

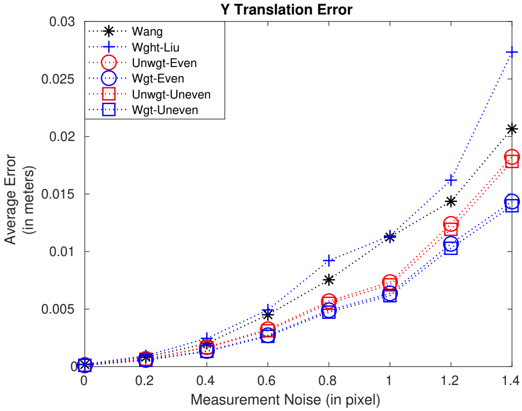


Figure 4.15: The translation error of **Y** of different methods and configurations with increased image noise.

Figure 4.14 and Figure 4.15 demonstrate the calibration errors. Several conclusions are derived as follows. Same as before, the weighted unevenly distributed dynamic pattern generates the most robust results. With the increase of the image noise, though both applying the weighted strategy and the configuration of the unevenly dynamic fiducial pattern generate increasing improvement, the contribution brought by the former is more significant since the imbalance of the measurement quality still exists in the dynamic fiducial pattern setup. The calibration performance of the weighted Liu's method and Wang's method is comparable, which verifies the improvement brought by the dynamic fiducial pattern without the final optimization could compete with the non-linear refinement based on the fixed-sized fiducial pattern.

4.6 Real Experiment Results

4.6.1 Experiment Setup

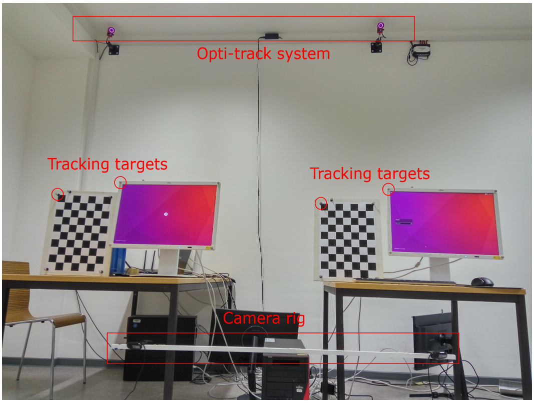


Figure 4.16: The calibration infrastructure in the real experiment. The checkerboards are of the same size as the monitors. The former is used for Liu's setup, and the monitors are used for the dynamic fiducial pattern setup.

Similar to the simulation, two setups, one for Wang's method and the dynamic fiducial pattern method, and the other for Liu's method, as well as two different image feature configurations, are performed in the real experiment. A camera rig with non-overlapping FOV and two 24-inch electronic displays with a resolution of 1900×1200 are introduced. Liu's setup consists of the same camera-rig and two rigidly connected fixed-sized calibration objects, which are of similar size as the electronic monitors (Figure 4.16). Same as before, both the decentralized and the centralized image feature distributions are implemented and compared. To validate the calibration results from different setups and configurations, a highly accurate tracking system ‘OptiTrack’ equipped in the calibration environment is used to provide the benchmark.

The implementation of Liu’s method is straightforward. The camera rig is placed at several different poses relative to the calibration rig, and a set of images $\{\mathbf{I}_i^{P1}, \mathbf{I}_i^{P2}\}_{i=1}^{i=n}$ containing the calibration patterns is collected for the relative pose estimation and the final refinement.

The complexity of the implementation of the dynamic fiducial pattern setup, when compared to Liu’s setup, is that the calibration pattern on the monitor is dynamically changing depending on the estimated relative pose between the camera and the corresponding monitor. In the simulation, the noise-corrupted pose estimation is applied, while in the real experiment, the estimated pose from the last timestamp is used to generate the current fiducial patterns. The camera frame rate in this experiment is set as 30 fps, and the movement of the camera rig is smooth and moderate, which ensures that applying the estimated pose from the last timestamp to generate the dynamic patterns for the current timestamp could be safely qualified. By changing the pose of the camera rig relative to the calibration rig, in this case, two monitors, a set of images $\{\mathbf{I}_i^{D1}, \mathbf{I}_i^{D2}\}_{i=1}^{i=n}$ containing the dynamic fiducial pattern, together with the well known 3D fiducial pattern information, is recorded.

In the simulation, the pose pair banks are carefully generated so that good-quality measurements and larger spatial distribution between the pose pairs are guaranteed, which is not feasible in the practical environment. Instead, a supervision program is additionally integrated into both setups in the real experiment. The program is an on-line implementation of the data selection strategy explained in the last chapter. The program first calculates the projection size of the fiducial pattern on the image, and only pose pairs that generate acceptable projection size are included, which guarantees the measurement quality. Meanwhile, the program examines the rotational and translational difference between the current pose pair and the gathered ones during the calibration procedure, which excludes the pose pair that lies in the neighborhood of the previously collected pose pairs. This helps reduce the potential instability by alleviating the spatial distribution imbalance of the collected pose pairs. Liu’s setup has to carefully balance the image quality against the spatial distribution level of the pose pairs since $\mathbf{A}_i$ and $\mathbf{B}_i$ are coupled by the closed-loop constraint $\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i$, which adds difficulty to obtain a proper set of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$. This inconvenience is alleviated in the dynamic fiducial pattern setup due to the improved measurement quality and the increased measurement space.

With the assistance of a highly accurate tracking system, the unknown transform $\mathbf{X}$ between the monitors could be accurately recovered after aligning the frame of the monitor to the frame of the tracking targets, which could be tracked by the ‘OptiTrack’ system. Two different configurations are applicable, namely the fixed trackable dynamic pattern setup and the unfixed trackable dynamic pattern setup. The difference lies in that the relative pose between the monitors is fixed in the former configuration while it could be adjusted in the latter one during the calibration procedure. The extra flexibility in the unfixed trackable dynamic pattern setup further improves the measurement quality since each calibration object could be independently placed to poses relative to the cameras with closer depth. The dynamic fiducial pattern is applied to both configurations to generate the most accurate relative pose.

In the fixed trackable dynamic pattern setup, a set of images $\{\mathbf{I}_i^{D1}, \mathbf{I}_i^{D2}\}_{i=1}^{i=n}$, the 3D coordinates of the corresponding dynamic patterns at each pose pair, and the recovered $\mathbf{X}$ using the tracking system are recorded and processed to run the weighted bundle adjustment formulated as follows:

$$(\hat{\mathbf{R}}_Y, \hat{\mathbf{t}}_Y) = \underset{\mathbf{R}_Y, \mathbf{t}_Y}{\arg\min} \sum_{i=1}^{n} (\lambda_i^{D1} \sum_{j=1}^{m(i)} \|\boldsymbol{\varepsilon}_{ij}^{D1}\|_2^2 + \lambda_i^{D2} \sum_{l=1}^{o(i)} \|\boldsymbol{\varepsilon}_{il}^{D2}\|_2^2),$$

where λ_i^{D1} and λ_i^{D2} are weighting factors.

In the unfixed trackable dynamic pattern setup, a set of images $\{\mathbf{I}_i^{D1}, \mathbf{I}_i^{D2}\}_{i=1}^{i=n}$, the corresponding dynamic pattern coordinates, and the recovered $\mathbf{X}_i$ applying the tracking system are recorded and processed to run the weighted BA similar to 4.6.1 except that the replacement of $\mathbf{A}_i$ and $\mathbf{B}_i$ under this circumstance becomes:

$$\begin{aligned} \mathbf{A}_i &= \mathbf{Y}\mathbf{B}_i\mathbf{X}_i^{-1}, \\ \mathbf{B}_i &= \mathbf{Y}^{-1}\mathbf{A}_i\mathbf{X}_i. \end{aligned}$$

The result of the weighted unfixed trackable dynamic pattern setup is served as the benchmark of $\mathbf{Y}$ since this is the most accurate estimation that could be generated in the real experiment.

4.6.2 Experimental Results

The results from Wang’s method, Liu’s method, the dynamic fiducial pattern method with two different image feature distributions, and the fixed trackable dynamic pattern method are compared against the benchmark, which is calculated from the weighted unfixed trackable dynamic pattern setup. The same error criteria are utilized as in the last chapter. Figure 4.17 demonstrates the results of different setups and configurations.

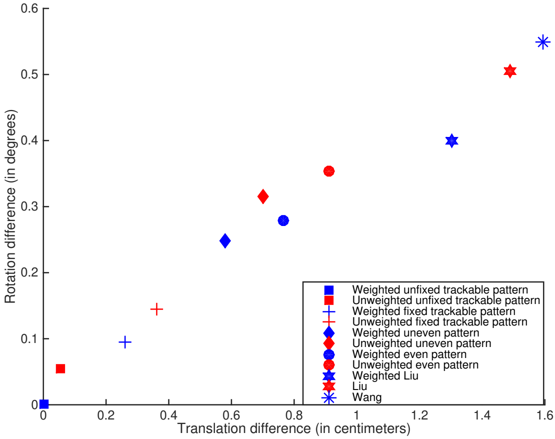


Figure 4.17: Calibration results of different setups and configurations. The calibration result from the weighted unfixed trackable dynamic pattern configuration is defined as the benchmark, which is located at the origin. The x-axis and y-axis show the corresponding translational and rotational deviations of different methods and configurations compared to the benchmark.

Most of the results from the real experiment are consistent with that in the simulation. The calibration results applying the weighting strategy deviate less from the ground truth regardless of different setups or configurations. Moreover, the more imbalanced the measurement quality is, the more improvement the weighting method will bring. All the setups applying either uneven or even dynamic fiducial patterns produce better results than the weighted Liu's method validating the improvement brought by the dynamic fiducial pattern compared to the fixed-sized calibration pattern. Besides, the uneven dynamic pattern produces less deviation from the benchmark compared to the even dynamic pattern.

However, the real experiment results show two conflicts compared to the results in the simulation. Firstly, the weighted evenly distributed dynamic pattern generates larger translational difference while smaller rotational difference compared to the unweighted unevenly distributed dynamic pattern, unlike in the simulation where the former always generates less calibration error. Second, the performance of Wang's method applying the unevenly distributed dynamic pattern is worse than the weighted Liu's method. In contrast, its performance is more or less the same compared to Liu's method in the simulation.

The above inconsistency may result because of the following reasons. First, the coordinate frame of the tracking targets might not be exactly aligned with the coordinate frame of the electronic screen due to the error introduced during the attachment of the targets. Second, though the tracking system is of high accuracy, it is unavoidable that the tracking targets could still be localized with a minor error, which in turn influences the calibration results.

4.7 Conclusions and Discussion

In this chapter, new eye-to-eye calibration methods applying fiducial dynamic patterns are proposed. The configuration of the virtual pattern displayed on the introduced monitors could be actively modified, which leads to larger pose change space and improved measurement quality. The improvement in the measurement space and the measurement quality helps alleviate the error propagation and the potential instability during the optimization process. Meanwhile, in contrast to the weighted Liu's method where numerical weighting factors are used, the dynamic fiducial pattern serves as a global non-numerical (structural) weighting factor, which normalizes the weight of each error item in the final objective function based on its inherent structure. Besides, the configuration of the introduced monitors could be easily readjusted for different camera rigs, so no customized calibration objects are needed. For the application where the camera rig is not movable, either additional equipment is included to assist the movement of the monitors, or a mobile camera rig needs to be introduced to pre-calibrate the relative pose between the monitors.

After applying the dynamic fiducial pattern to the eye-to-eye calibration like in this thesis, both the accuracy of the single pose estimation $(\mathbf{A}_i, \mathbf{B}_i)$ and the calculation of the unknown extrinsics $\mathbf{X}, \mathbf{Y}$ have been improved. In the future, the integration of the dynamic pattern concept to other applicable situations will be explored, such as multi-robot cooperative localization, active perception-action for a mobile agent, the assistance of efficient quadrotor landing, etc[1].

[1] In Chapter 6, a specific robot team is presented, which integrates the application of the dynamic pattern.

5 Cooperative Localization Methods: MOMA and S-MOMA

Most of the cooperative localization research focuses on the heterogeneous robot team, where the robots are equipped with different kinds of sensors. The positioning of the robots is resolved after fusing different sources of measurement data based on their quality. In the first chapter, the advantages of CRL and the difficulty of localizing robots within the indoor environment with repetitive features and a deficient number of features are explained. Depending on whether environmental measurements are applied or not to influence the localization results, CRL methods are classified into two categories, and typical methods in each category are concisely reviewed. In this chapter, two CRL methods are presented: MOMA and S-MOMA, which could be applied to a variety of different situations, including the indoor environment, where the localization methods are prone to failure. The former could be considered as a particular version of VO, which uses fiducial features instead of environmental features. The latter is developed based on MOMA and fits the framework of V-SLAM, where a global map is built in order to reduce the accumulating drift. Both MOMA and S-MOMA do not need to transplant artificial markers to the environment, and they are purely vision-based approaches since cameras are the only sensor used for localization.

Despite different localization methods of VO, BA is always applied for estimating the camera pose $(\mathbf{R}(t), \mathbf{t}(t))$. The objective function of BA is formulated as:

$$(\hat{\mathbf{R}}(t), \hat{\mathbf{t}}(t), \hat{\mathbf{P}}_1, \hat{\mathbf{P}}_2 \cdots \hat{\mathbf{P}}_n) = \underset{\mathbf{R}(t), \mathbf{t}(t), \mathbf{P}_1, \mathbf{P}_2 \cdots \mathbf{P}_n}{\arg\min} \sum_{i=1}^{n} \left\| \mathbf{p}_i - \frac{1}{\lambda_i} \mathbf{K} \left(\mathbf{R}(t) \mathbf{P}_i + \mathbf{t}(t) \right) \right\|_2^2 .$$

The main error of VO stems from the noise in 2D measurements $\mathbf{p}_i$, which introduces unavoidable error to both camera pose and 3D feature landmarks $\mathbf{P}_i$. The error then propagates to the consecutive estimates, which leads to the accumulating drift. Besides, if only a monocular camera is used, the scale of the environment will also drift since one camera is not adequate to recover the true scale of the scene.

There are several ways to reduce the error in the VO framework. First, additional sensors such as inertial measurement units (IMUs) or a navigation system, for instance, a global positioning system (GPS) can be integrated to alleviate the drift and scale problem. A stereo camera could be used instead to recover the accurate scale of the environment and prevent scale drift. Some of the VO methods also build a local map. In contrast, it is preferable in V-SLAM to build a globally consistent map, especially in an enclosed space. The global map is used to recognize loop closure, where the previous features are re-detected and included to run a global BA to alleviate the drift. In general, the map-building procedure is prone to error due to non-static features, ambiguous features, and erroneous 2D-2D correspondence, which are highly dependent on the environment.

In this thesis, MOMA aims at reducing the localization error by providing unambiguously detectable, error-free fiducial landmarks. In this case, the robust features $\mathbf{P}_i$ used to estimate camera pose are known a priori and do not contain any error. Thus, there is no error in $\mathbf{P}_i$ propagating to the following estimates, unlike the methods which use the estimated $\hat{\mathbf{P}}_i$ which contains unavoidable errors. S-MOMA improves the map-building process by additionally fusing introduced fiducial features that help provide robust relative pose and prevent erroneous data associations.

5.1 MOMA

MOMA extends the work presented in [31] by replacing the expensive laser sensor with a cheap monocular camera.

For simplicity, the cooperative mechanism of MOMA is explained using a group of two robots, named Apollo (A) and Boreas (B) respectively. However, the proposed cooperative localization approach could be generalized to any vision-based multi-robot system that has more than two robot members. In the two-robot configuration, a planar fiducial marker is attached to the Boreas's backside, based on which the accurate and robust relative pose between Apollo and the marker board is recovered. The movement pattern of the robot team, which resembles the movement of the caterpillar, is demonstrated in Figure 5.1. Both robots are motionless at the starting point, then one robot, for example, Boreas, moves first while the other robot stays stationary. Then Boreas stops, and Apollo is driven to the predefined position relative to Boreas. The above procedure repeats until both robots reach the target position.

This recursive positioning process is similar to most CRL methods [31], [1], [15], and the basics of the algorithm are formulated in Table 5.1.

The explanation of some representations in Table 5.1 is given, which helps to ease the difficulty for understanding what follows. The coordinate frame of

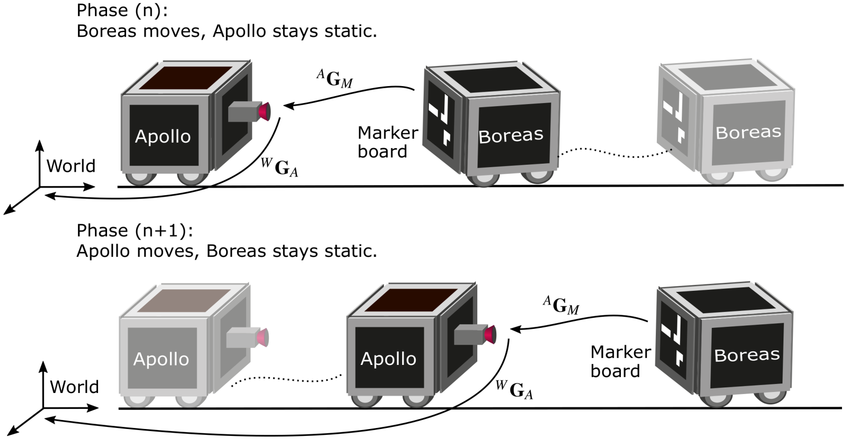


Figure 5.1: The caterpillar-like movement pattern of MOMA. As shown in the figure, only a monocular camera is adequate. Apollo is mounted with a monocular camera, and a fiducial marker is attached to the backside of Boreas. Boreas moves first, and during this phase Apollo keeps static. Then Boreas stops, and Apollo is driven to the predefined position relative to Boreas. These phases repeat until both reach the destination.

Apollo is defined as its monocular camera frame, and the coordinate frame of Boreas is aligned with the system frame of its fiducial board. All the transformations in the table are denoted in a similar form as ${}^{W}\mathbf{G}_{A}(t)$[1], which describes the relationship from frame A to frame W at time t. The phase N is defined as the period of time during which the state of the robots keeps unchanged. At each phase N, the robots have two movement states, namely static and mobile. During each phase, at least one robot needs to be static in order to serve as the beacon for the rest of the group members. At phase N, the pose of the static robot, for example, in this case, Apollo, is denoted as ${}^{W}\mathbf{G}_{A}(t_n^s)$, where the superscript s indicates the robot is static. The relative pose between the mobile Boreas and the static Apollo is described as ${}^{A}\mathbf{G}_{M}(t_n^k)$, where the subscript M indicates marker frame and the superscript k represents the current timestamp at phase N. At the end of phase N, Boreas will stop, and both robots are static. The relative pose of Boreas to Apollo is depicted as ${}^{A}\mathbf{G}_{M}(t_n^s)$. Again, the superscript s here has two indications. First, the previously moving robot Boreas is now static. Second, this is the end of the phase, when the robots' state exchange happens.

[1] It is a 4×4 matrix in the form of $\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$ representing the special Euclidean transformation.

Table 5.1: Recursive localization procedure of MOMA.

Phase 0: starting point. ${}^W\mathbf{G}_A(t_0) = \mathit{Identity}$
Phase I: Boreas moves, Apollo is static. ${}^W\mathbf{G}_A(t_1^s) = {}^W\mathbf{G}_A(t_0) \quad \mathit{Remains\ static}$
Phase II: Apollo moves, Boreas is static. ${}^W\mathbf{G}_A(t_2^k) = {}^W\mathbf{G}_A(t_0){}^A\mathbf{G}_M(t_1^s)({}^A\mathbf{G}_M(t_2^k))^{-1}$
...
Even Phase ($N = 2\tau, \tau \in \mathbb{N}$): Apollo moves, Boreas is static. ${}^W\mathbf{G}_A(t_n^k) = {}^W\mathbf{G}_A(t_{n-1}^s){}^A\mathbf{G}_M(t_{n-1}^s)({}^A\mathbf{G}_M(t_n^k))^{-1}$
Odd Phase ($N = 2\tau + 1, \tau \in \mathbb{N}$): Boreas moves, Apollo is static. ${}^W\mathbf{G}_A(t_n^s) = {}^W\mathbf{G}_A(t_{n-1}^s) \quad \mathit{Remains\ static}$
...

Though MOMA does not contain any error in 3D features, the error accumulates in a different way. Expanding the formula at even phase $N = 2\tau, \tau \in \mathbb{N}$ in Table 5.1 when Apollo moves leads to:

$${}^W\mathbf{G}_A(t_n^k) = {}^W\mathbf{G}_A(t_0){}^A\mathbf{G}_M(t_1^s)...{}^A\mathbf{G}_M(t_{n-3}^s){}^M\mathbf{G}_A(t_{n-2}^s)\,{}^A\mathbf{G}_M(t_{n-1}^s)({}^A\mathbf{G}_M(t_n^k))^{-1}.$$

The camera pose estimate is a cascade of the current relative pose ${}^A\mathbf{G}_M(t_n^k)$ and the poses at all exchange states which occur at the end of each phase. All these poses are estimated using fiducial landmarks and contain an error. However, compared to VO, MOMA accumulates error at a more discrete timestamp since only the exchange-state poses are included. Besides, the camera pose estimate is more robust and accurate due to the introduced fiducial features.

5.2 S-MOMA

In this section, the cooperative localization method S-MOMA is presented. The method localizes a group of robots by fusing pose estimates from static environmental features and mobile fiducial features. The cooperation mechanism between the robots is first explained, followed by the formulation of the cooperative localization algorithm.

5.2.1 Cooperative Mechanism

A group of two robots, namely Apollo and Boreas, is used to explain the algorithm. Same as MOMA, Apollo is equipped with a monocular camera for recovering the relative pose by detecting the fiducial board, which is mounted on the backside of Boreas. By building a global map, the accumulating drift could be reduced, especially when the robot returns to previously visited places. Therefore, the front side of Boreas is installed with a stereo camera in S-MOMA, which is used to build a global map of the environment. With the above configuration, S-MOMA is able to inherit essential characteristics from MOMA while at the same time evolves by interacting with its surroundings.

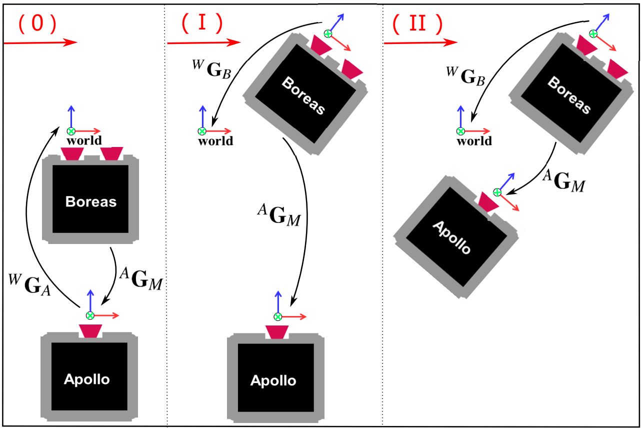


Figure 5.2: Caterpillar-like movement pattern of S-MOMA. As shown in the first column, the fixed world reference frame is aligned with the left camera coordinate frame of Boreas before both robots start to move. Boreas moves first, and during this phase Apollo keeps static. Then Boreas stops, and Apollo is driven to the predefined relative position of Boreas. These phases repeat until both reach the destination.

The movement pattern of S-MOMA demonstrated in Figure 5.2 is very similar to MOMA. However, the role that the robot pair plays in S-MOMA is not the same as in MOMA [1]. The difference lies in the designed functionalities of the equipped cameras. Same as MOMA, Apollo's camera detects the fiducial marker board rigidly attached to Boreas and generates relative pose measurements. In this case, a monocular camera is enough. While in S-MOMA, the stereo camera pair mounted on Boreas is used for performing a vision-based localization algorithm using static landmarks from the environment. Though each robot in the team may perform both relative pose detection and self-localization during the localization procedure, which provides more flexibility to the path planning of each robot, the functionality of the two-robot group keeps unchanged during the deduction of S-MOMA as well as in the following experiment section for consistency and better comparison, which means: the relative pose detection is solely performed by the monocular camera mounted on Apollo; only Boreas could localize itself independently using its onboard stereo camera.

Table 5.2: Recursive localization procedure of S-MOMA.

Phase 0: starting point. ${}^W\mathbf{G}_B(t_0) = \mathit{Identity}$ ${}^W\mathbf{G}_A(t_0) = {}^W\mathbf{G}_B(t_0){}^B\mathbf{G}_M{}^M\mathbf{G}_A(t_0)$
Phase I: Boreas moves, Apollo is static. ${}^W\mathbf{G}_A(t_1^s) = {}^W\mathbf{G}_A(t_0)$ *Remains static* ${}^W\mathbf{G}_B(t_1^k) = {}^W\mathbf{G}_A(t_1^s){}^A\mathbf{G}_M(t_1^k)({}^B\mathbf{G}_M)^{-1}$
Phase II: Apollo moves, Boreas is static. ${}^W\mathbf{G}_B(t_2^s) = {}^W\mathbf{G}_A(t_1^s){}^A\mathbf{G}_M(t_1^s)({}^B\mathbf{G}_M)^{-1}$ *Remains static* ${}^W\mathbf{G}_A(t_2^k) = {}^W\mathbf{G}_B(t_2^s){}^B\mathbf{G}_M{}^M\mathbf{G}_A(t_2^k)$
...
Even Phase ($N = 2\tau, \tau \in \mathbb{N}$): Apollo moves, Boreas is static. ${}^W\mathbf{G}_B(t_n^s) = {}^W\mathbf{G}_A(t_n^s){}^A\mathbf{G}_M(t_n^s)({}^B\mathbf{G}_M)^{-1}$ *Remains static* ${}^W\mathbf{G}_A(t_n^k) = {}^W\mathbf{G}_B(t_n^s){}^B\mathbf{G}_M{}^M\mathbf{G}_A(t_n^k)$
Odd Phase ($N = 2\tau + 1, \tau \in \mathbb{N}$): Boreas moves, Apollo is static. ${}^W\mathbf{G}_A(t_n^s) = {}^W\mathbf{G}_B(t_{n-1}^s){}^B\mathbf{G}_M{}^M\mathbf{G}_A(t_{n-1}^s)$ *Remains static* ${}^W\mathbf{G}_B(t_n^k) = {}^W\mathbf{G}_A(t_n^s){}^A\mathbf{G}_M(t_n^k)({}^B\mathbf{G}_M)^{-1}$
...

The resulting localization algorithm based on relative pose estimation is presented in Table 5.2. The representation in S-MOMA follows the tradition of MOMA. The coordinate frame of Apollo is defined as its monocular camera frame, which is the same as MOMA, while the coordinate system of Boreas is defined as its left camera frame. ${}^W\mathbf{G}_B(t_n^k)$ describes the transform from Boreas camera frame to the world reference frame at timestamp k during phase N, and the superscript W stands for the world coordinate system, which is aligned with the left camera frame of Boreas at the starting point. ${}^B\mathbf{G}_M$, which needs to be estimated, is denoted as the transform from Boreas marker board frame to its left camera system, where the subscript M indicates marker.

Same as MOMA, the localization error also accumulates in S-MOMA. Expanding the formula at odd phases $N = 2\tau + 1, \tau \in \mathbb{N}$ in Table 5.2 when Boreas moves leads to:

$$^{W}\mathbf{G}_B(t_n^k) = {}^{B}\mathbf{G}_M{}^{M}\mathbf{G}_A(t_0){}^{A}\mathbf{G}_M(t_1^s)...{}^{M}\mathbf{G}_A(t_{n-1}^s){}^{A}\mathbf{G}_M(t_n^k)({}^{B}\mathbf{G}_M)^{-1},$$

in between are concatenated transformations from previous phases, which could be simplified as follows:

$$^{W}\mathbf{G}_B(t_n^k) = {}^{B}\mathbf{G}_M{}^{M_0}\mathbf{G}_{M_n}(t_n^k)({}^{B}\mathbf{G}_M)^{-1},$$

in which ${}^{M_0}\mathbf{G}_{M_n}(t_n^k)$ depicts the transform from the current marker frame to the marker frame at the starting point.

The above formulas reveal two important indications when Boreas moves. First, the transformation ${}^{B}\mathbf{G}_M$, which is a pre-calculated measurement with unavoidable calibration error, is always included in the equation. However, the calibration error does not accumulate with increased state exchanges. Second, with more state exchanges during the localization process, more noise-corrupted measurements are introduced and become dominant over time, which causes vision-based CRL error-drift in the long run. However, the benefits that the system could gain from those methods are high robustness and accuracy of the recovered relative pose. First, the 3-D coordinates of the fiducial marker are exactly known compared to the triangulated 3-D feature points in the environment, which contain an error. Second, the 3D-2D feature correspondence is much more robust when using the fiducial marker, while matching features from the environment is inherently less reliable. So even though vision-based CRL methods are error-drift, they could provide more robust and accurate relative pose estimation.

5.2.2 Eye-to-marker Calibration

In vision-based CRL methods where fiducial markers are attached to the robot, recovering the accurate transformation between the onboard camera and the fiducial board is a prerequisite for executing other tasks. Such a configuration is not uncommon. For example, in S-MOMA, the transformation ${}^{B}\mathbf{G}_M$ between the left camera and the fiducial board has to be accurately estimated since static environmental features and mobile fiducial features are going to be fused.

When the fiducial marker appears in the FOV of the camera, the transformation could be directly recovered. Otherwise, an external camera and an external

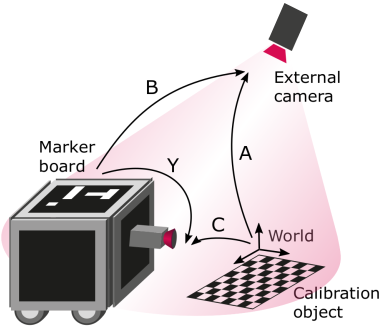


Figure 5.3: The demonstration of eye-to-marker calibration. In this configuration, the onboard marker and the external calibration object are placed within the FOV of the corresponding camera.

calibration pattern are introduced, like in Figure 5.3. In order that the unknown $\mathbf{Y}$ could be recovered from solving the closed-loop $\mathbf{Y} = \mathbf{C}\mathbf{B}^{-1}\mathbf{A}$, where $\mathbf{C}$, $\mathbf{B}$, and $\mathbf{A}$ are the relative pose between camera and fiducial marker, the configuration has to meet the following conditions. First, the external planar pattern and the onboard marker have to be in the FOV of the external camera. Meanwhile, the external calibration object has to appear in the FOV of the robot camera.

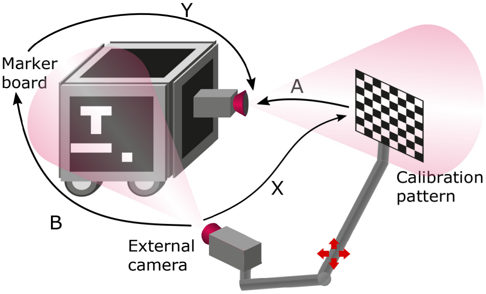


Figure 5.4: The demonstration of eye-to-marker calibration. In this configuration, the calibration device, which is rigidly linked with an external camera and a planar calibration pattern, is introduced to relate the onboard camera and marker.

However, the above requirements cannot always be fulfilled. When the external camera and the external calibration object could not relate to the onboard robot camera and marker, the above method could not be applied. This situation is similar to eye-to-eye calibration and is defined as eye-to-marker calibration following the tradition. In this case, a calibration device is constructed by rigidly linking a planar fiducial pattern and an external camera (Figure 5.4). The calibration device is placed in several different positions to the onboard camera and marker, where each marker could be captured by the corresponding camera. In the end, a set of pose pairs $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{i=n}$ is collected, and the initial estimation of $\mathbf{X}$ and $\mathbf{Y}$ is calculated by solving $\{\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i\}_{i=1}^{i=n}$. Same as eye-to-eye calibration, the initial value is then applied to minimize the reprojection error based objective function in order to improve calibration accuracy.

5.2.3 Fusion Strategy and Optimization Method

Roumeliotis et al. validated in [54] that continuous relative pose detection would bring in more accuracy than intermittent relative observation. As mentioned before, map building within indoor environments where repetitive localization is frequently required alleviates drift error, which compensates the weakness of CRL methods. However, the ambiguity property of environment features and their diverse distribution grant map construction a desirable but thorny task. Fortunately, the mapping task could be less challenging when provided with extra reference to robust and accurate relative pose estimates. The fusing strategy is based on combining the benefits of indoor map building with continuous relative pose detection. The caterpillar-like movement pattern described above makes the best use of the concept.

The on-line frame-by-frame fusion happens only during the movement of the ‘explorer’, which could localize itself using the equipped stereo camera as well as receive the relative pose from its group member. In this case, the so-called explorer is Boreas. During the movement of Boreas, two sources of information are simultaneously generated at each timestamp. First, Boreas localizes itself by applying V-SLAM based algorithm, which enables localization as well as map building. Meanwhile, Apollo which now acts as a portable landmark provides reliable relative positioning. This localization coupling introduces extra constraints and could be optimized for further refinement.

The proposed objective function combines two different objectives: (1) sum of reprojection errors ε_{env} of triangulated environment feature points and (2) sum of reprojection errors ε_{marker} of the exactly known marker corners. The subscripts *env* and *marker* differentiate the reprojection error generated from environment and marker. The main difference between ε_{env} and ε_{marker} is: due to the discreteness property of vision sensors and unavoidable sensor noise, the triangulated 3-D environment feature points $\mathbf{X}_i$ from 2-D images contain undesired uncertainty. In contrast, the 3-D coordinates $\mathbf{M}_i$ of the fiducial markers are noise-free, which increases the robustness of the localization system after including them into the pose-decision process.

In this thesis, the coordinate frame of Boreas is defined as its left camera frame. At time step t, the pose of Boreas represented in world frame W is denoted as ${}^{W}\mathbf{G}_{B}(t)$, which is going to be estimated. These two objective functions are first formulated separately.

On the one hand, ${}^{W}\mathbf{G}_{B}(t)$ could be calculated by minimizing the reprojection errors ε_{env}, defined as:

$$ {}^{W}\hat{\mathbf{G}}_{B}(t) = \underset{{}^{W}\mathbf{G}_{B}(t)}{\arg\min} \sum_{i=1}^{n(t)} \left(\varepsilon_{env}^{i}(t)\right)^{2}, \tag{5.1} $$

$$ \varepsilon_{env}^{i}(t) = \left\|\mathbf{x}_{i}(t) - \boldsymbol{\pi}\left({}^{W}\mathbf{G}_{B}(t)\mathbf{X}_{i}(t)\right)\right\|_{2}, \tag{5.2} $$

where $\mathbf{X}_{i}(t) = \left(X_{i}(t), Y_{i}(t), Z_{i}(t), 1\right)$ represents the homogeneous coordinates of triangulated 3-D feature points and $\mathbf{x}_{i}(t) = \left(x_{i}(t), y_{i}(t), 1\right)$ the corresponding 2-D homogeneous coordinates at time t. The $\boldsymbol{\pi}$ appeared in equation (5.2) is a 3×4 projection matrix, which projects 3-D feature points onto the 2-D image plane. Equation (5.2) gives the reprojection error of each triangulated feature point detected at the current frame, and by minimizing the sum, ${}^{W}\mathbf{G}_{B}(t)$ is estimated.

The way how ${}^{W}\mathbf{G}_{B}(t)$ is estimated by applying equation (5.1) and (5.2) is analogous to the majority of VO and V-SLAM algorithms.

On the other hand, ${}^{W}\mathbf{G}_{B}(t)$ could also be estimated by minimizing the reprojection errors ε_{marker}, specified as:

$$ {}^{A}\hat{\mathbf{G}}_{M}(t) = \underset{{}^{A}\mathbf{G}_{M}(t)}{\arg\min} \sum_{j=1}^{m(t)} \left(\varepsilon_{marker}^{j}(t)\right)^{2}, \tag{5.3} $$

$$ \varepsilon_{marker}^{j}(t) = \left\|\mathbf{m}_{j}(t) - \boldsymbol{\pi}\left({}^{A}\mathbf{G}_{M}(t)\mathbf{M}_{j}\right)\right\|_{2}, $$

and

$$ {}^{W}\hat{\mathbf{G}}_{B}(t) = {}^{W}\mathbf{G}_{A}(t){}^{A}\hat{\mathbf{G}}_{M}(t)\left({}^{B}\mathbf{G}_{M}\right)^{-1}, \tag{5.4} $$

where $\mathbf{M}_{j} = (X_{j}, Y_{j}, Z_{j}, 1)$ represents exactly known 3-D fiducial features that are time-independent with regard to marker coordinate frame, and $\mathbf{m}_{j}(t) = \left(x_{j}(t), y_{j}(t), 1\right)$ are the corresponding 2-D image coordinates. Same as before, $\boldsymbol{\pi}$ represents the projection matrix. Since in this phase Boreas is moving, Apollo is static, so ${}^{W}\mathbf{G}_{A}(t)$ is phase-constant and could be deduced from previous measurements (refer to Table 5.2). ${}^{A}\mathbf{G}_{M}(t)$ denotes the transformation between Apollo and the marker board mounted on Boreas, and ${}^{B}\mathbf{G}_M$ describes the relationship between Boreas's marker board and its own frame, which has been well calibrated beforehand. The relationship between these transforms is illustrated in Figure 5.5.

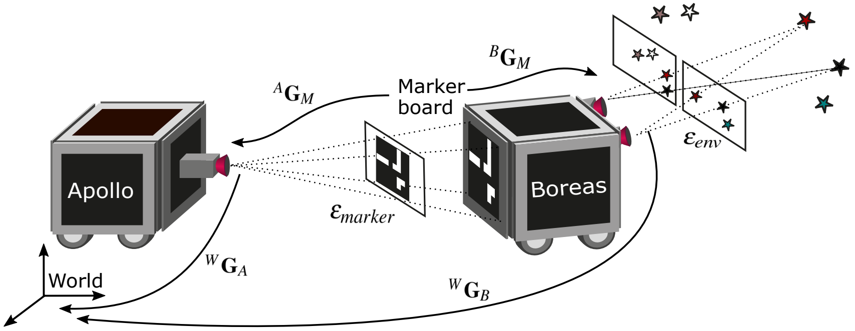


Figure 5.5: The illustration of the fusion strategy. The proposed objective function consists of the sum of reprojection errors caused by triangulated environment feature points and the sum of reprojection errors generated from fiducial marker corners, whose 3-D coordinates are accurately known. The fusion happens during the movement of Boreas.

Minimizing (5.3) results in an estimated ${}^{A}\hat{\mathbf{G}}_M(t)$, which creates optimally estimated ${}^{W}\hat{\mathbf{G}}_B(t)$ after applying equation (5.4). Most localization algorithms which are not environment interactive apply this rule.

As stated before, the robot team should be viewed as an entire entity. So in order to achieve global optimization of ${}^{W}\mathbf{G}_B(t)$, the reprojection errors ε_{env} and ε_{marker} also need to be considered as a whole. They are bonded and fused in one objective function formulated as follows:

$$ {}^{W}\hat{\mathbf{G}}_B(t) = \underset{{}^{W}\mathbf{G}_B(t)}{\arg\min} \left\{ \frac{1}{n(t)} \sum_{i=1}^{n(t)} \left(\varepsilon_{env}^{i}(t)\right)^2 + \frac{\lambda}{m(t)} \sum_{j=1}^{m(t)} \left(\varepsilon_{marker}^{j}(t)\right)^2 \right\}. \quad (5.5) $$

The environment generally has many more feature points compared with the implanted fiducial features. In order to counteract the imbalance of the number of different feature sources, the reprojection errors are normalized by their corresponding feature numbers. Considering the measurement quality difference mentioned before, an additional influence factor λ is introduced to stress the value of accurate fiducial features. In the end, what the objective function minimizes is the sum of weighted-average reprojection errors of those two feature sources. One reasonable criterion of configuring λ is the quality difference between the cameras, which are used for V-SLAM and relative pose detection. The influence of environmental features and fiducial features behaves in an on-line manner since real-time feature detection is running at each frame.

The non-linear function (5.5) needs a reliable initial value of ${}^{W}\mathbf{G}_{B}(t)$ at each timestamp. For robustness consideration, the localization result based on the relative pose estimate from the current frame will be applied. This guarantees the robustness even when the environment has deficient or ambiguous features. The former would cause the system to crash while the latter introduces disastrous outliers under the V-SLAM framework.

This fusion scheme described above is illustrated in Figure 5.5. By minimizing the objective function (5.5), ${}^{W}\mathbf{G}_{B}(t)$ is updated using the optimum of the objective, which can be calculated as fast as the on-line frame rate.

5.3 Experiments and Evaluation

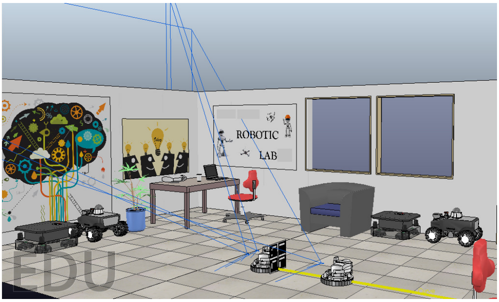


Figure 5.6: The simulation scene constructed in V-REP.

Different experiments are implemented in simulation using V-REP. The simulated environment is an imitated version of a robot laboratory (Figure 5.6), where dynamic robot models, vision sensors, fiducial markers, and common indoor items are constructed. The robot model being tested is Robotino from the Festo company. Same as before, Boreas is equipped with a pair of stereo camera and a rigidly attached Aruco marker board [47] on its backside for relative pose detection, and Apollo is mounted with an HD camera. The procedure of camera intrinsic calibration and extrinsic calibration, such as the transformation between the camera and the marker board, is rigorously conducted in simulation beforehand.

As explained before, the two robots move in turn. During the movement of

Boreas, the algorithm estimates its pose by fusing the measurement data from the stereo camera with the relative measurements provided by Apollo. When Apollo moves, its pose could be recovered simply by concatenating its relative pose to the positioning of Boreas. One criterion used in the experiment for deciding when one robot should stop and the other should move (state exchange) is the reprojection error of the detected marker. When the reprojection error is larger than the defined threshold, the moving robot stops, and the other starts to move, which ensures the accuracy of the vision-based system.

The proposed algorithm S-MOMA has been implemented under the ROS framework, and the code is available online [2]. The SLAM framework that S-MOMA builds on is S-PTAM [3]. S-PTAM, referred to as stereo parallel tracking and mapping, is a well-known and highly recognized V-SLAM based algorithm [51].

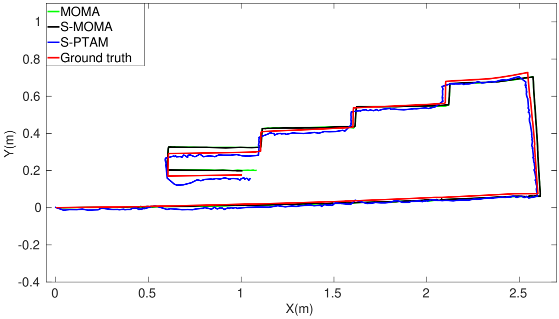


Figure 5.7: Trajectories of different localization methods.

MOMA, S-MOMA, as well as S-PTAM, are implemented in three different testing environments. For each experiment conducted, the same measurement data are processed, applying those three localization methods.

First, the implementation of those methods is carried out in an environment filled with rich, distinct, and evenly distributed features. Figure 5.7 demonstrates four trajectories of the localization results of MOMA, S-MOMA, S-PTAM, and the recorded ground truth from one test. The trajectories differentiate each other using different colors. The corresponding localization error compared to ground truth is shown in Figure 5.8. In this test, **S-MOMA** generates the least localization error of $\mathbf{0.74}cm$ compared to S-PTAM of $1.03cm$, and MOMA of $1.45cm$,

[2] https://github.com/zaijuan/Cooperative-Localization

[3] https://github.com/lrse/sptam

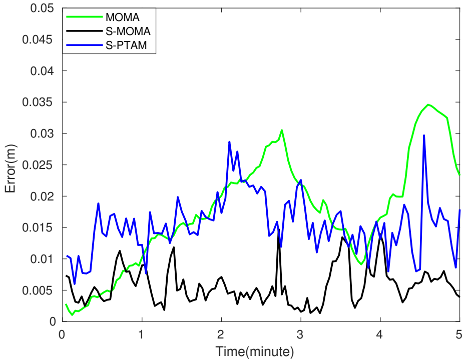


Figure 5.8: Localization errors regarding different methods.

and the average error percentages with regard to the trajectory length are 0.24% for MOMA, **0.12**% for S-MOMA, and 0.17% for S-PTAM.

In Figure 5.8, it is noticeable that MOMA shows increasing error accumulation. However, it is worth mentioning that in the beginning, the localization accuracy of MOMA is very competitive. In most cases, S-PTAM shows less drift compared to MOMA due to the constructed map, while its localization error is still larger when compared to S-MOMA. The reason lies in that S-MOMA generates more accurate and robust positioning along the way due to the fusion mechanism, which further assists in building a more accurate map of the environment. So when the robot returns to its previously visited places, S-MOMA, which constructs a better map, generates less localization error.

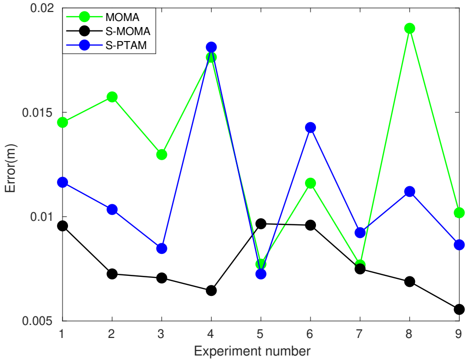


Figure 5.9: Localization errors of different methods from 9 tests.

Another nine experiments are conducted with about $6m$ long trajectory, and the overall results are shown in Figure 5.9. Except for experiment 5, which shows slightly worse results, all the other experiments verify that S-MOMA outperforms MOMA and S-PTAM with more accurate localization results. The average errors of MOMA, **S-MOMA**, and S-PTAM from all experiments are $1.31cm$, $\mathbf{0.77}cm$, and $1.10cm$, and the corresponding average error percentages with regard to the trajectory length are 0.21%, $\mathbf{0.13}\%$, and 0.18%.

We implement those localization methods in the second environment with a deficient number of features. It turns out that S-PTAM gets stuck at the initialization step, while S-MOMA behaves the same as MOMA since the number of environmental features, in this case, is almost none.

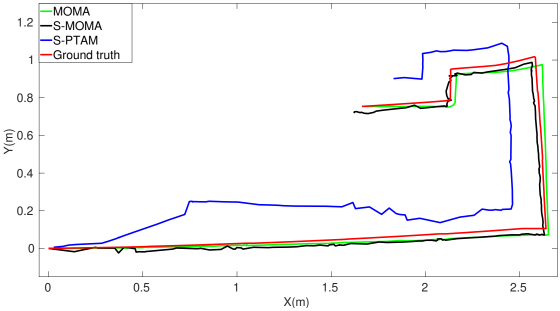


Figure 5.10: Trajectories of different methods with ambiguous features.

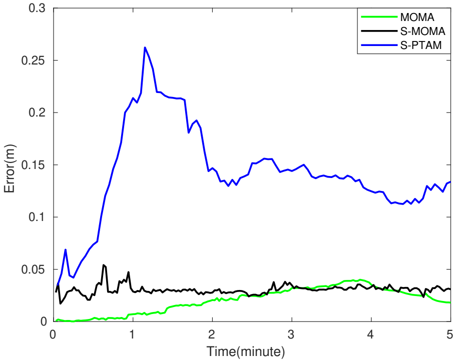


Figure 5.11: Localization errors of different methods with ambiguous features.

In the end, the performance of those localization methods in environments filled with ambiguous features is investigated. Therefore, another experiment is conducted in a simulation scene with deliberately designed ambiguous features by adding symmetric items such as a chessboard, chair, etc. The testing results generated from this configuration are illustrated in Figure 5.10 and Figure 5.11.

The trajectory of S-PTAM deviates wildly from the ground truth since ambiguous features unavoidably introduce mismatches even with the help of motion prediction and RANSAC outlier rejection mechanism. In this case, the localization results from S-MOMA are also influenced. The average errors of MOMA, **S-MOMA**, and S-PTAM are $1.34cm$, $\mathbf{1.62}cm$ and $14.42cm$, and the corresponding average error percentages with regard to the trajectory length are 0.24%, $\mathbf{0.28}\%$, 2.5%. It shows that S-MOMA could still demonstrate almost the same level of accuracy compared to MOMA but with much more robustness.

The results from different experiments have verified that S-MOMA generates the least localization error under various testing environments. The reason behind this deserves a more in-depth analysis. When the environment is full of rich and good-quality features, S-MOMA is co-biased by measurements from the environment as well as the mobile fiducial markers, and the pose estimation is updated to the optimum after frame-rate optimization. Just like in SLAM, accurate localization results lead to an accurate and robust map building, which further benefits the localization itself when the robot could sense the constructed map. This is why S-MOMA outperforms the V-SLAM approach due to this coupled effect of mapping building and positioning. Because of the inherently increased drift error in MOMA, S-MOMA also demonstrates better results in the long run. Moreover, the longer the trajectory is, the more advantages S-MOMA shows. While the environment has fewer or ambiguous features, S-MOMA is robust enough to prevent the system from getting trapped in unstable estimations and calculates the positioning by relying more on relative measurements accordingly. This explains the improved performance when compared to a V-SLAM framework.

S-MOMA behaves like a wise decision-maker. It understands the quality of different measurements, learns the environment being placed in, and is aware of how to fuse different feature sources based on that. In the end, S-MOMA manifests satisfactory accuracy and robustness under various testing circumstances. Based on all experimental results, S-MOMA has been validated to resolve all the difficulties discussed in the introduction chapter.

5.4 Conclusions and Discussion

In this section, two localization methods, namely MOMA and S-MOMA, are proposed. MOMA realizes cooperative VO by introducing fiducial markers attached to other robots. The localization algorithm concatenates all state-exchange poses which occur at the end of each phase. By fusing the pose estimates from environmental features and relative pose estimates from mobile fiducial marker features, S-MOMA preserves the essence of MOMA and further improves the localization robustness and accuracy under various testing environments. Both MOMA and S-MOMA are practical, versatile and could be generalized into any suitable platform, where it is possible to fuse the absolute pose estimate with the relative pose estimate based on the same error type.

However, the configuration of the robot team and the cooperation mechanism implemented in this thesis are simplified. To add more ‘vitality’ to the algorithm, possible improvements are as follows. First, during the localization process, the ‘explorer’ could actively grab preferable measurements from the environment to potentially improve the accuracy as long as it is in the reliable detection range of its colleagues. Second, the functionality of the robots could dynamically vary according to their surroundings or the demands from other team members instead of being fixed, which enables more flexibility and efficiency to the system. Even though these variations in configuration and cooperation mechanism introduce additional overheads and increase management complexity, continuous improvement on localization robustness, accuracy, flexibility as well as efficiency would be brought to the multi-robot system.

6 Conclusion

6.1 Summary

This thesis contributes to two research fields: eye-to-eye calibration and cooperative robot localization (CRL).

6.1.1 Eye-to-eye Calibration

In this thesis, a weighted non-linear optimization method and a data selection strategy are proposed to alleviate the underlying instabilities in Liu's calibration method. The optimization method introduces an extra quality measure factor in the objective function based on the projection size of the calibration object, which relieves the measurement imbalance, improves calibration accuracy, and increases robustness against noise. Besides, by carefully choosing a subset from the collected pose pairs, the possibility of getting trapped in a local minimum is reduced. The optimization method and the data selection strategy are applicable to any calibration setup, which minimizes the sum of reprojection errors and is constrained by the closed-loop pose transformations $\mathbf{AX} = \mathbf{YB}$.

The proposed optimization method and the data selection strategy do not increase the measurement space or improve the measurement quality. Instead, they choose a subset and passively weigh each pose pair based on the measurement quality. Therefore, a new calibration method applying a highly accurate tracking system is proposed, which disconnects the rigid link in Liu's setup. Thus, the calibration patterns could be independently placed in front of the camera pair. The method eliminates the instabilities in Liu's setup and shows high accuracy.

However, the cost of introducing a highly accurate tracking system is usually expensive. Inspired by the widespread application of electronic displays, another new calibration method applying the fiducial pattern is introduced. By a proper encoding of the fiducial patterns on display, the method increases the pose change space. The configuration of the virtual pattern displayed on the introduced monitors is actively modified to generate better-quality measurements, which helps to reduce the error propagation and the potential instability during the optimization process. The dynamic pattern provides a cascade of dual functionalities: a local non-numerical weighting for each single pose estimation and a global non-numerical weighting for the overall objective function. Instead of using numerical weighting factors to evaluate the measurement quality like the optimization method applied in Liu's method, the dynamic fiducial pattern serves as a non-numerical, structural weighting factor, which normalizes the weight of each error item in the final objective function based on its inherent structure. Besides, the configuration of the introduced monitors is flexible since they could be easily re-adjusted for different camera rigs, so no customized calibration objects are needed.

6.1.2 Cooperative Robot Localization

Two CRL methods are proposed: MOMA and S-MOMA.

MOMA is a VO method using the mobile fiducial markers attached to the other robots. Thus no marker intervention is introduced to the environment. Compared to typical VO methods where the environmental features and the estimated camera pose are coupled and correlated, MOMA uses fiducial landmarks to recover the robot pose. Besides, the drift in MOMA occurs at discrete timestamps when the exchange-state poses are incorporated and concatenated.

In order to further reduce the drift in MOMA, S-MOMA fuses the pose estimates from environmental features and the relative pose estimates from mobile fiducial marker features. Thus, a robust, accurate, and globally consistent map is constructed. Compared to V-SLAM methods that are prone to failure when the environment has ambiguous, repetitive features or a deficient number of features, S-MOMA demonstrates high accuracy and robustness under various testing environments due to the fusion of the robust, accurate relative pose estimates. The algorithm is practical, versatile, and could be generalized into any suitable platform, wherever it is possible to fuse the absolute pose estimate with the relative pose estimate based on the same error type.

6.2 Outlook

Though extracting measurement sets with good-quality from the pose pair bank is convenient in the simulation, the generation of pose pair bank based on ground truth is time-consuming and not practical in real experiments. Without a particular assistance, the data collection procedure in Liu's setup is challenging. Considering the underlying twisted and limited measurement space, an on-line program that interactively assists the collection of the measurements in real time could be developed, such that a better balance between the spatial distribution of collected pose pairs and their measurement quality could be perceptively kept.

In the dynamic fiducial pattern setup, the generated patterns are corners, which could be replaced or combined with other feature types to increase the detection accuracy, such as circles. In the simulation, the intrinsic parameters of the camera are assumed to be well-calibrated, and in the real experiment, they are pre-calibrated before the extrinsic camera calibration. The combination of the intrinsic calibration and the extrinsic calibration makes the whole calibration procedure more efficient, automatic, and accurate. Besides, the application of the dynamic fiducial patterns should not be constrained only to eye-to-eye calibration. The integration of the dynamic pattern concept to other applicable situations is worth exploring, such as multi-robot cooperative localization, active perception-action for a mobile agent, the assistance of efficient quadrotor landing, etc.

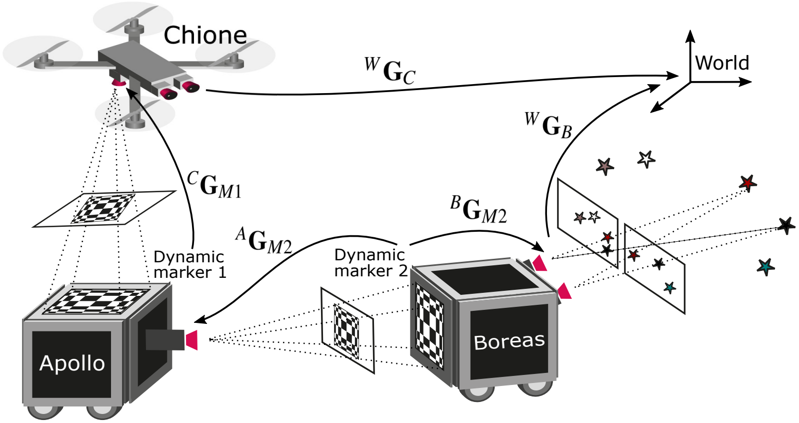


Figure 6.1: Demonstration of a cooperative robot localization configuration using a group of three robots. The top of Apollo and the back of Boreas are equipped with electronic monitors for actively displaying dynamic fiducial patterns. All the robots could be actively controlled or passively detected with high accuracy by the dynamic fiducial pattern.

In MOMA and S-MOMA, the configuration of the robot team and the cooperation mechanism that they have been applied to are simplified. The prospect of introducing dynamic fiducial patterns to cooperative robot localization within the indoor environment is promising. The active measurements generated from the dynamic fiducial patterns could be applied to assist, and bias the estimation of relevant states effectively, such as the robot pose or the coordinates of the environmental features. For example, in Figure. 6.1, two mobile robots (Apollo, Boreas) and a quadrotor (Chione) forms a group of three robots. The monitor screen on the back of Boreas could actively display fiducial patterns to accurately recover the relative pose to Apollo or control its movement. Meanwhile, the electronic monitor on top of Apollo could be used to actively control Chione and passively detect its pose relative to Apollo. In this case, all the robots could be related to each other by different constraints. The activeness in terms of the generation of dynamic fiducial pattern and control of the robots introduces more certainty and robustness to the system, while the passiveness allows the system to perceive and interact with its environment accurately. Even though these variations in configuration and cooperation mechanism introduce additional overheads and increase management complexity, continuous improvement on localization robustness, accuracy, flexibility as well as efficiency would be brought to the multi-robot system.

Bibliography

[1] Raul Acuna, Zaijuan Li, and Volker Willert. Moma: Visual mobile marker odometry. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 206–212. IEEE, 2018.

[2] Raul Acuna, Robin Ziegler, and Volker Willert. Single pose camera calibration using a curved display screen. *Forum Bildverarbeitung*, pages 25–36, 2018.

[3] Raul Acuña and Volker Willert. Insights into the robustness of control point configurations for homography and planar pose estimation. *arXiv preprint arXiv:1803.03025*, 2018.

[4] Erdinç Altuğ, James P. Ostrowski, and Camillo J. Taylor. Control of a quadrotor helicopter using dual camera visual feedback. *The International Journal of Robotics Research*, 24(5):329–341, 2005.

[5] Esra Ataer-Cansizoglu, Yuichi Taguchi, Srikumar Ramalingam, and Yohei Miki. Calibration of non-overlapping cameras using an external slam system. In *Proceedings of the 2nd International Conference on 3D Vision*, volume 1, pages 509–516. IEEE, 2014.

[6] Timothy D. Barfoot. *State Estimation for Robotics*. Cambridge University Press, 2017.

[7] Serge Belongie. Rodrigues' rotation formula. *From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld. wolfram. com/RodriguesRotationFormula. html*, 1999.

[8] Fernando Gomez Bravo, Alberto Vale, and Maria Isabel Ribeiro. Particle-filter approach and motion strategy for cooperative localization. In *Proceedings of the Internacional Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 2006.

[9] Vincenzo Caglioti, Augusto Citterio, and Andrea Fossati. Cooperative, distributed localization in multi-robot systems: a minimum-entropy approach.

In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS)*, pages 25–30. IEEE, 2006.

[10] Gerardo Carrera, Adrien Angeli, and Andrew J. Davison. Slam-based automatic extrinsic calibration of a multi-camera rig. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2652–2659. IEEE, 2011.

[11] Jianhui Cheng, Shunan Ren, Guolei Wang, Xiangdong Yang, and Ken Chen. Calibration and compensation to large-scale multi-robot motion platform using laser tracker. In *Proceedings of the IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 163–168. IEEE, 2015.

[12] Louis G. Clift and Adrian F. Clark. Determining positions and distances using collaborative robots. In *Proceedings of the 7th International Conference on Computer Science and Electronic Engineering Conference (CEEC)*, pages 189–194. IEEE, 2015.

[13] José Alexandre De França, Marcelo Ricardo Stemmer, Maria Bernadete De M. França, and Juliani Chico Piai. A new robust algorithmic for multi-camera calibration with a 1d object under general motions without prior knowledge of any camera intrinsic parameter. *Pattern Recognition*, 45(10):3636–3647, 2012.

[14] Konstantinos G. Derpanis. The harris corner detector. *York University*, 2004.

[15] Vikas Dhiman, Julian Ryde, and Jason J Corso. Mutual localization: Two camera relative 6-dof pose estimation from reciprocal fiducial observation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1347–1354. IEEE, 2013.

[16] Jing Dong, Erik Nelson, Vadim Indelman, Nathan Michael, and Frank Dellaert. Distributed real-time cooperative localization and mapping using an uncertainty-aware expectation maximization approach. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5807–5814. IEEE, 2015.

[17] Fadi Dornaika and Radu Horaud. Simultaneous robot-world and hand-eye calibration. *IEEE Transactions on Robotics and Automation*, 14(4):617–622, 1998.

[18] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics & Automation Magazine*, 13(2):99–110, 2006.

[19] Sandro Esquivel, Felix Woelk, and Reinhard Koch. Calibration of a multi-camera rig from non-overlapping views. In *Joint Pattern Recognition Symposium*, pages 82–91. Springer, 2007.

[20] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8(3):325–344, 2000.

[21] Xiaoshan Gao, Xiaorong Hou, Jianliang Tang, and Hangfei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):930–943, 2003.

[22] Zheng Gong, Zhen Liu, and Guangjun Zhang. Flexible global calibration of multiple cameras with nonoverlapping fields of view using circular targets. *Applied Optics*, 56(11):3122–3131, 2017.

[23] Robert Grabowski, Luis E. Navarro-Serment, Christiaan J.J. Paredis, and Pradeep K. Khosla. Heterogeneous teams of modular robots for mapping and exploration. *Autonomous Robots*, 8(3):293–308, 2000.

[24] Adam Harmat, Michael Trentini, and Inna Sharf. Multi-camera tracking and mapping for unmanned aerial vehicles in unstructured environments. *Journal of Intelligent & Robotic Systems*, 78(2):291–317, 2015.

[25] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[26] Lionel Heng, Mathias Bürki, Gim Hee Lee, Paul Furgale, Roland Siegwart, and Marc Pollefeys. Infrastructure-based calibration of a multi-camera rig. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4912–4919. IEEE, 2014.

[27] Lionel Heng, Bo Li, and Marc Pollefeys. Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1793–1800. IEEE, 2013.

[28] Radu Horaud and Fadi Dornaika. Hand-eye calibration. *The International Journal of Robotics Research*, 14(3):195–210, 1995.

[29] Michael Kaess and Frank Dellaert. Probabilistic structure matching for visual slam with a multi-camera rig. *Computer Vision and Image Understanding*, 114(2):286–296, 2010.

[30] Ram Krishan Kumar, Adrian Ilie, Jan-Michael Frahm, and Marc Pollefeys. Simple calibration of non-overlapping cameras with a mirror. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–7. IEEE, 2008.

[31] Ryo Kurazume, Shigemi Nagata, and Shigeo Hirose. Cooperative positioning with multiple robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1250–1257. IEEE, 1994.

[32] Pierre Lébraly, Clément Deymier, Omar Ait-Aider, Eric Royer, and Michel Dhome. Flexible extrinsic calibration of non-overlapping cameras using a planar mirror: Application to vision-based robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5640–5647. IEEE, 2010.

[33] Pierre Lébraly, Eric Royer, Omar Ait-Aider, Clément Deymier, and Michel Dhome. Fast calibration of embedded non-overlapping cameras. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 221–227. IEEE, 2011.

[34] Gim Hee Lee, Friedrich Fraundorfer, and Marc Pollefeys. Structureless pose-graph loop-closure with a multi-camera system on a self-driving car. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 564–571. IEEE, 2013.

[35] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal of Computer Vision*, 81(2):155, 2009.

[36] Aiguo Li, Lin Wang, and Defeng Wu. Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product. *International Journal of Physical Sciences*, 5(10):1530–1536, 2010.

[37] Y. Liu, J.R. Lin, T. Liu, C. Liu, and S. Ye. Multi-sensor global calibration technology of vision sensor in car body-in-white visual measurement system. *Acta Meteorologica Sinica*, 5:204–209, 2014.

[38] Zhen Liu, Fengjiao Li, and Guangjun Zhang. An external parameter calibration method for multiple cameras based on laser rangefinder. *Measurement*, 47:954–962, 2014.

[39] Zhen Liu, Guangjun Zhang, Zhenzhong Wei, and Junhua Sun. A global calibration method for multiple vision sensors based on multiple targets. *Measurement Science and Technology*, 22(12):125102, 2011.

[40] Zhen Liu, Guangjun Zhang, Zhenzhong Wei, and Junhua Sun. Novel calibration method for non-overlapping multiple vision sensors based on 1d target. *Optics and Lasers in Engineering*, 49(4):570–577, 2011.

[41] Zhibin Liu, Mingguo Zhao, Zongying Shi, and Wenli Xu. Multi-robot cooperative localization through collaborative visual object tracking. In *Robot Soccer World Cup*, pages 41–52. Springer, 2007.

[42] Cristina Losada, Manuel Mazo, Sira Palazuelos, Daniel Pizarro, and Marta Marrón. Multi-camera sensor system for 3d segmentation and localization of multiple mobile robots. *Sensors*, 10(4):3261–3279, 2010.

[43] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An invitation to 3-D vision: from images to geometric models*, volume 26. Springer Science & Business Media, 2012.

[44] Raj Madhavan, Kingsley Fregene, and Lynne E Parker. Distributed cooperative outdoor multirobot localization and mapping. *Autonomous Robots*, 17(1):23–39, 2004.

[45] Luis Montesano, José Gaspar, José Santos-Victor, and Luis Montano. Cooperative localization by fusing vision-based bearing measurements and motion. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2333–2338. IEEE, 2005.

[46] Jorge J. Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical Analysis*, pages 105–116. Springer, 1978.

[47] Rafael Munoz-Salinas. Aruco: a minimal library for augmented reality applications based on opencv. *Universidad de Córdoba*, 2012.

[48] Suraj Nair, Giorgio Panin, Martin Wojtczyk, Claus Lenz, Thomas Friedelhuber, and Alois Knoll. A multi-camera person tracking system for robotic applications in virtual reality tv studio. In *Proceedings of the 17th IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[49] David Nistér, Oleg Naroditsky, and James Bergen. Visual odometry. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–I. Ieee, 2004.

[50] Benjamin Petit, Jean-Denis Lesage, Clément Menier, Jérémie Allard, Jean-Sébastien Franco, Bruno Raffin, Edmond Boyer, and François Faure. Multicamera real-time 3d modeling for telepresence and remote collaboration. *International Journal of Digital Multimedia Broadcasting*, 2010, 2010.

[51] Taihú Pire, Thomas Fischer, Javier Civera, Pablo De Cristóforis, and Julio Jacobo Berlles. Stereo parallel tracking and mapping for robot localization. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1373–1378. IEEE, 2015.

[52] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, pages 430–443. Springer, 2006.

[53] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2010.

[54] Stergios I. Roumeliotis and George A. Bekey. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*, 18(5):781–795, 2002.

[55] Sajad Saeedi, Liam Paull, Michael Trentini, and Howard Li. Multiple robot simultaneous localization and mapping. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 853–858. IEEE, 2011.

[56] Mili Shah. Solving the robot-world/hand-eye calibration problem using the kronecker product. *Journal of Mechanisms and Robotics*, 5(3):031007, 2013.

[57] Rajnikant Sharma, Stephen Quebe, Randal W. Beard, and Clark N. Taylor. Bearing-only cooperative localization. *Journal of Intelligent & Robotic Systems*, 72(3-4):429–440, 2013.

[58] Stephen M. Smith and J. Michael Brady. Susan—a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.

[59] Joan Solà, Jeremie Deray, and Dinesh Atchuthan. A micro lie theory for state estimation in robotics. *arXiv preprint arXiv:1812.01537*, 2018.

[60] Hauke Strasdat, José M.M. Montiel, and Andrew J. Davison. Visual slam: why filter? *Image and Vision Computing*, 30(2):65–77, 2012.

[61] Tobias Strauß, Julius Ziegler, and Johannes Beck. Calibrating multiple cameras with non-overlapping views using coded checkerboard targets. In *Proceedings of the 17th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 2623–2628. IEEE, 2014.

[62] Junhua Sun, Huabin He, and Debing Zeng. Global calibration of multiple cameras based on sphere targets. *Sensors*, 16(1):77, 2016.

[63] Yasuhiro Suzuki, Masato Koyamaishi, Tomohiro Yendo, Toshiaki Fujii, and Masayuki Tanimoto. Parking assistance using multi-camera infrastructure. In *Intelligent Vehicles Symposium (IV)*, pages 106–111. IEEE, 2005.

[64] Amy Tabb and Khalil M. Ahmad Yousef. Parameterizations for reducing camera reprojection error for robot-world hand-eye calibration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3030–3037. IEEE, 2015.

[65] Michael J. Tribou, Adam Harmat, David W.L. Wang, Inna Sharf, and Steven L. Waslander. Multi-camera parallel tracking and mapping with non-overlapping fields of view. *The International Journal of Robotics Research*, 34(12):1480–1500, 2015.

[66] Stephen Tully, George Kantor, and Howie Choset. Leap-frog path design for multi-robot cooperative localization. In *Field and Service Robotics*, pages 307–317. Springer, 2010.

[67] Veeravalli S. Varadarajan. *Lie groups, Lie algebras, and their representations*, volume 102. Springer Science & Business Media, 2013.

[68] Jiaole Wang, Liao Wu, Max Q-H Meng, and Hongliang Ren. Towards simultaneous coordinate calibrations for cooperative multiple robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 410–415. IEEE, 2014.

[69] Robert WM Wedderburn. Quasi-likelihood functions, generalized linear models, and the gauss—newton method. *Biometrika*, 61(3):439–447, 1974.

[70] Patrick Wunsch, Stefan Winkler, and Gerd Hirzinger. Real-time pose estimation of 3d objects from camera images using neural networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 4, pages 3232–3237. IEEE, 1997.

[71] Renbo Xia, Maobang Hu, Jibin Zhao, Songlin Chen, Yueling Chen, and ShengPeng Fu. Global calibration of non-overlapping cameras: state of the art. *Optik-International Journal for Light and Electron Optics*, 158:951–961, 2018.

[72] Meng Xie, Zhenzhong Wei, Guangjun Zhang, and Xinguo Wei. A flexible technique for calibrating relative position and orientation of two cameras with no-overlapping fov. *Measurement*, 46(1):34–44, 2013.

[73] Hanqi Zhuang, Zvi S. Roth, and Raghavan Sudhakar. Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form ax=yb. *IEEE Transactions on Robotics and Automation*, 10(4):549–554, 1994.