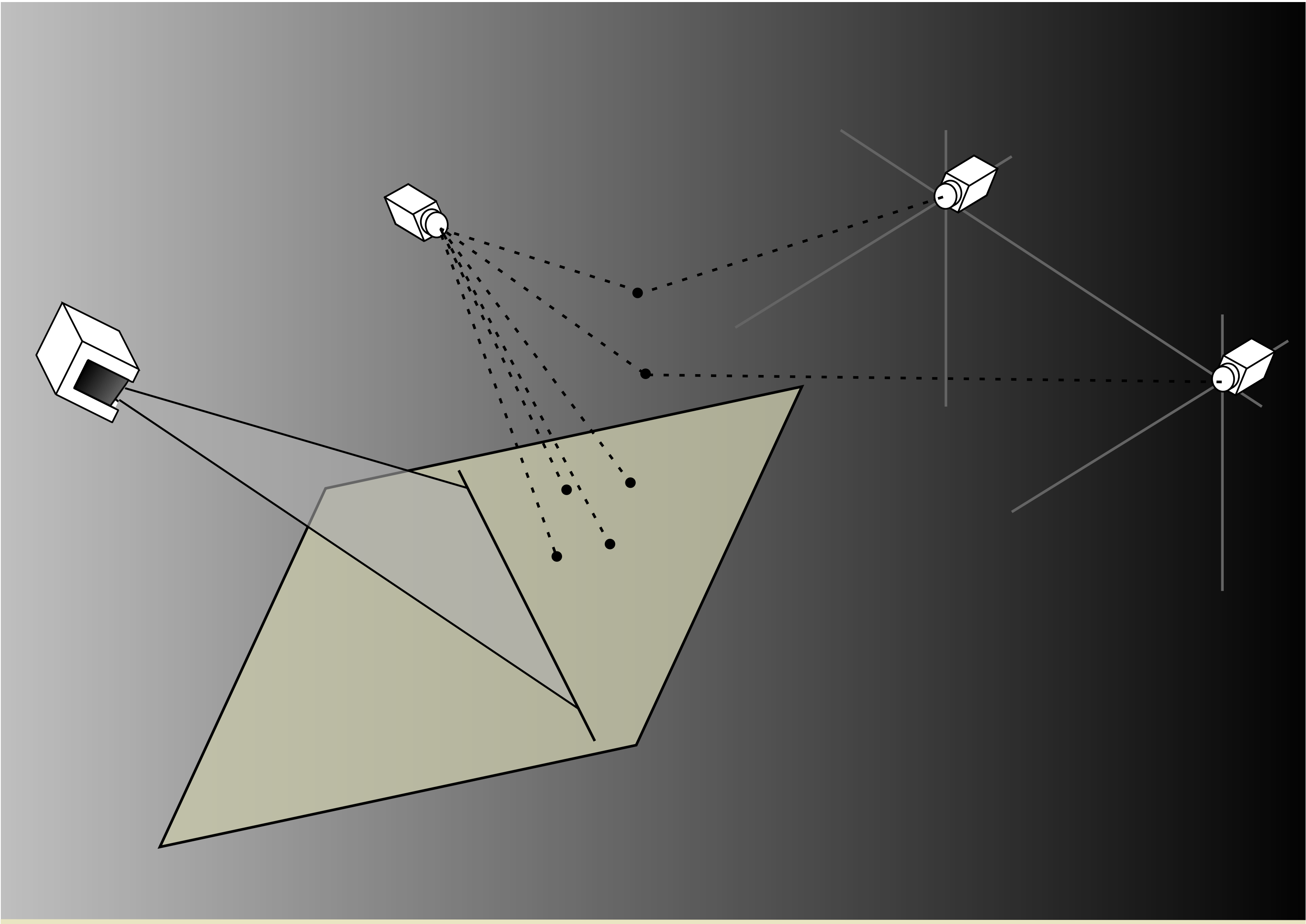


Francisco Porto Guerra e Vasconcelos

MINIMAL SOLUTIONS TO GEOMETRIC PROBLEMS WITH MUTLIPLE CAMERAS OR MULTIPLE SENSOR MODALITIES

Tese de doutoramento em Engenharia Electrotécnica e de Computadores, Ramo de Especialização em Automação e Robótica, orientada pelo Senhor Professor Doutor João Pedro Barreto e apresentada ao Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Fevereiro de 2015

• U C •

Universidade de Coimbra

· U C ·

FCTUC FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

PhD Thesis

Minimal Solutions to Geometric Problems with Multiple Cameras or Multiple Sensor Modalities

Francisco Vasconcelos

February 27, 2015

Minimal Solutions to Geometric Problems with Multiple Cameras or Multiple Sensor Modalities

By

Francisco Vasconcelos

Advisor: Prof. Dr. João P. Barreto

Department of Electrical and Computer Engineering
Faculty of Science and Technology,
University of Coimbra

February 27, 2015

Contents

Acknowledgements

First of all I want to thank my advisor João Barreto for giving me the opportunity to join a great research team and fully supporting my work as a PhD student. Without his commitment and patience this thesis would not be possible.

I thank all my colleagues, specially Miguel Lourenço, Michel Antunes, Rui Melo, Luís Santos and Carolina Raposo that helped me uncountable times and provided a great environment in the lab.

I also thank Edmond Boyer for giving me the opportunity to stay at INRIA–Grenoble for three months, contributing with valuable input to this thesis, as well as the rest of the MORPHEO team for their support and friendly environment.

I acknowledge the Portuguese Science Foundation (FCT) for funding my PhD program through grant SFRH/BD/72323/2010, and the Institute for Systems and Robotics – Coimbra for supporting my research.

List of Tables

List of Figures

Abstract

This thesis addresses minimal problems that involve multiple cameras or a combination of cameras with other sensors, particularly focusing on four cases: extrinsic calibration between a camera and a laser rangefinder (LRF); full calibration of an ultrasound array (US) with a camera; full calibration of a camera within a calibrated network; relative pose between axial systems.

The first problem (LRF-Camera) is highly important in the context of mobile robotics in order to fuse the information of an LRF and a Camera in localization maps. The second problem (US-Camera) is becoming increasingly relevant in the context of medical imaging to perform guided intervention and 3D reconstruction with US probes. Both these problems use a planar calibration target to obtain a minimal solution from 3 and 4 correspondences respectively. They are formulated as the registration between planes detected by the camera and lines detected by either the LRF or the US.

The third problem (Camera-Network) is concerned with two application scenarios: addition of a new camera to a calibrated network, and tracking of a hand-held camera within the field of view of a calibrated network. The last problem (Axial System) has its main application in motion estimation of stereo camera pairs. Both these problems introduce a 5-dimensional linear subspace to model line incidence relations of an axial system, of which a pair of calibrated cameras is a particular example. In the Camera-Network problem a generalized fundamental matrix is derived to obtain a 11-correspondence minimal solution. In the Axial System problem a generalized essential matrix is derived to obtain a 10-correspondence non-minimal solution. Although it should be possible to solve this last problem with as few as 6 correspondences, the proposed solution is the closest to minimal in the literature.

Additionally this thesis addresses the use of the RANSAC framework in the context of the problems mentioned above. While RANSAC is the most widely used method in computer vision for robust estimation when minimal solutions are available, it cannot be applied directly to some of the problems discussed here. A new framework – multiset-RANSAC – is presented as an adaptation of RANSAC to problems with multiple sampling datasets. Problems with multiple cameras or multiple sensors often fall in this category and thus this new framework can greatly improve their results. Its applicability is demonstrated in both the US-Camera and the Camera-Network problems.

Resumo

Esta tese aborda os *problemas mínimos* no contexto de visão por computador, isto é, problemas com o mesmo número de restrições e de parâmetros desconhecidos, para os quais existe um conjunto finito e discreto de soluções. A tese foca-se em particular nos seguintes problemas: calibração extrínseca entre uma câmara e um sensor laser *rangefinder* (LRF); calibração completa de uma sonda ultrasom (US) com uma câmara; calibração completa de uma câmara dentro de uma rede calibrada; estimação de pose relativa entre sistema axiais.

O primeiro problema (*LRF-Camera*) é extremamente importante no contexto de robótica móvel para fundir a informação de um sensor LRF e uma câmara em mapas de localização. O segundo problema (*US-Camera*) está-se a tornar cada vez mais relevante no contexto de imagiologia médica para realizar intervenções guiadas e reconstrução 3D com sondas ecográficas. Ambos os problemas usam um alvo de calibração planar para obter uma solução mínima usando 3 e 4 correspondências respectivamente, e são formulados como o registo 3D entre planos detectados pela câmara e linhas detectadas pelo LRF ou US.

O terceiro problema (*Camera-Network*) tem duas aplicações em mente: a introdução de uma nova câmara numa rede calibrada, e o seguimento de uma câmara guiada manualmente dentro do campo de visão de uma rede calibrada. O último problema (*Axial System*) tem a sua maior aplicação na estimação de pose relativa entre pares de câmaras estéreo. Em ambos os problemas é introduzido um subespaço linear em 5 dimensões que modela as relações de incidência de linhas num sistema axial, do qual as câmaras estéreo são um caso particular. No problema *Camera-Network* é introduzida uma generalização da matriz fundamental que permite obter uma solução mínima com 11 correspondências. No problema *Axial System* é introduzida uma generalização da matrix essencial que permite obter uma solução não mínima com 10 correspondências. Apesar de ser possível, em teoria, resolver este último problema com apenas 6 correspondências, a solução apresentada nesta tese usa um menor número de correspondências que as alternativas existentes.

Adicionalmente esta tese aborda o uso de RANSAC no contexto dos problemas anteriormente descritos. O RANSAC é o estimador robusto mais utilizado em visão por computador quando existem soluções mínimas para um determinado problema, no entanto não pode ser aplicado directamente em algumas das aplicações aqui descritas. Um novo método é proposto – multiset-RANSAC – que adapta o RANSAC para situações que envolvem a amostragem de múltiplos conjuntos de dados. Os problemas com múltiplas câmaras ou múltiplos sensores encontram-se mutas vezes nesta categoria, tornando o multiset-RANSAC numa ferramenta que pode melhorar bastante os resultados em alguns dos problemas focados nesta tese. A utilidade deste método é demonstrada nos problemas *US-Camera* e *Camera-Network*.

Chapter 1

Introduction

A solution to a problem with N unknowns is called minimal if it is determined from exactly N constraints. Although in practice most geometric problems in computer vision deal with over-determined problems, minimal solutions are important to fully characterize all the constraints involved in a problem and improve other estimation methods. Minimal solutions can be used to enforce hard constraints in the presence of noisy data, e.g., to guarantee that a rotation matrix is orthonormal ($\mathsf{R}^{\mathsf{T}}\mathsf{R} = \mathtt{I}$) or a fundamental matrix is rank deficient ($\det \mathsf{F} = 0$). Non-minimal solutions require an over-parameterization (e.g. rotation represented by it 9 matrix parameters) that does not guarantee the end result belongs to the solution subspace, and thus an additional projection step must be performed. Minimal solutions are also useful to perform robust estimation with RANSAC [1] in datasets contaminated by outliers. These problems generally use feature correspondences of the same geometric entity under different viewpoints to generate constraints [2]. Feature detection and matching algorithms are prone to errors and therefore some of these correspondences are outliers. RANSAC iteratively generates solutions by sampling random sets of correspondences until there is enough evidence that the best solution is generated only from inlier correspondences. It can be demonstrated that the lowest number of RANSAC iterations is achieved by generating solutions from the lowest possible number of random samples and therefore minimal solutions maximize the efficiency of this estimator [1]. A more detailed review of RANSAC can be found on the next chapter.

In most cases, minimal problems boil down to solving polynomial systems. However, until recently it was very hard to efficiently solve complex polynomial systems. Early minimal algorithms were limited to relatively simple problems which only require to solve either linear systems, such as the 4-point homography estimation [3], or very simple polynomial systems, such as the p3p problem [4]. In more complex cases, some approaches avoid polynomial systems by using non-minimal data to cast the problem linearly [5, 6].

Polynomial solvers have recently seen a great improvement and as a result a huge number of solutions to minimal problems have been proposed over the last years. In a first development, some polynomial systems have been solved by either carefully casting the problem and engineering a solution [7] or using resultant-based methods [8]. While each of these alternatives cannot be readily generalized to other problems, the most recent contributions provide a systematic framework to solve a wide range of polynomial systems in an efficient way. A first approach developed a method to build polynomial solvers by numerically computing a Groebner Basis to a given polynomial system [9]. Later approaches avoid the computation of Groebner bases and provide a systematic way to recast polynomial systems as eigenvalue problems [10, 11].

The current state-of-the-art in minimal solutions for computer vision problems [12] mainly addresses problems involving a single moving camera, including:

- Relative pose [7, 8] and absolute orientation [13] of calibrated cameras
- Relative pose [9] and absolute orientation [14] of cameras with unknown focal length
- Relative pose [15] and absolute orientation [14] of cameras with unknown radial distortion
- Relative pose [16] and absolute orientation [17] of non-central systems.

In this thesis we are interested in minimal solutions to problems that, unlike the cases mentioned above, either involve multiple cameras or a combination of cameras with other sensor modalities. This includes a much wider variety of problems that have not been solved yet. In the past few years there have been an increasing interest in multi-sensor systems. Under some circumstances, the minimal problems involving non-central systems are equivalent to systems of multiple calibrated cameras [16,18]. Some minimal solutions were also proposed to the relative pose between stereo camera pairs under different configurations of correspondences [19, 20]. Regarding the combination with other sensor modalities, some examples are the minimal solution for relative pose [21] and absolute orientation [22] when the additional information of an IMU sensor is available, and also the extrinsic calibration between a camera and an RGB-D sensor [23].

In this thesis we contribute to the following problems:

- A minimal solution to the extrinsic calibration between a camera and a laser-range finder. This is a widely used set-up in automotive and robotics applications.

- A minimal solution to the extrinsic calibration between a camera and an ultrasound array sensor. This set-up is being increasingly used in medicine in the context of interventional imaging.

- A minimal solution to the intrinsic and extrinsic calibration of a camera using pairwise correspondences with a calibrated camera network. This problem is useful in the context of automatically inserting a new node into a camera network or tracking a moving camera within the field of view of a camera network.

- A non-minimal solution to the relative pose of an axial system using independent pairwise correspondences. Although in this case we do not provide a minimal solution, it is the closest to minimal in literature. An axial camera model can be applied, e.g., to stereo camera setups.

- A RANSAC framework for multiple dataset sampling – multiset-RANSAC, addressing the required adaptations to the RANSAC-family algorithms when candidate solutions are obtained by sampling multiple datasets. This problem arises in most geometric problems involving multiple cameras or multiple sensor modalities.

1.1 Thesis Organization

In **Chapter 2** we describe the multiset-RANSAC formulation, that is further used to validate the solutions to some of the geometric problems presented in this thesis. This is unpublished material to be included in a future submission.

Each of the following 4 Chapters focuses a different geometric problem:

Chapter 3 Extrinsic calibration between a camera and a laser range-finder. It corresponds to the contributions in [24] with some additional simplifications to the formulation.

Chapter 4 Extrinsic calibration between a camera and an ultra-sound array. This is unpublished material to be included in a future submission.

Chapter 5 Extrinsic and intrinsic calibration of a camera given pairwise correspondences with a camera network. It corresponds to the contributions in [25], with an extended set of unpublished results to be included in a future submission.

Chapter 6 Relative pose between calibrated axial cameras. It corresponds to the contributions in [26].

Note that the formulation used on **Chapter 4** builds on results from **Chapter 3**, and the formulation used on **Chapter 6** builds on results from **Chapter 5**.

Finally, **Chapter 7** discusses the overall conclusions of this thesis.

1.2 Notation

Scalars are represented by plain letters, e.g. λ, vectors by bold symbols, e.g. $\mathbf{t}$, matrices by letters in sans serif font, e.g. T, sets by letters in mathcal font, e.g. $\mathcal{S}$.

3D lines are expressed in homogeneous Plücker coordinates, e.g. the 6×1 vector $\mathbf{L}$.

The equality up to scale is denoted by $\sim$ in order to be distinguished from the strict equality $=$.

The operator $[\mathbf{v}]_\times$ designates the 3×3 skew symmetric matrix of a 3×1 vector $\mathbf{v}$.

The operator $\otimes$ designates the kronecker product.

Single valued superscripts, e.g. $\mathsf{T}^{\{n\}}$, are used to denote the nth column of matrix T. Additionally we use $\mathsf{T}^{\{x:y,w:z\}}$ to denote a submatrix of T that contains the elements ranging from row x to y and column w to z.

All plot distributions are done with the Matlab function *boxplot* that shows the two middle quartiles of the distribution (25th to 75th percentiles) as a box with a horizontal line at the median. The whisker edges refer to the lowest and highest quartiles, and the crosses show data beyond 1.5 times the interquartile range (outliers in the distribution).

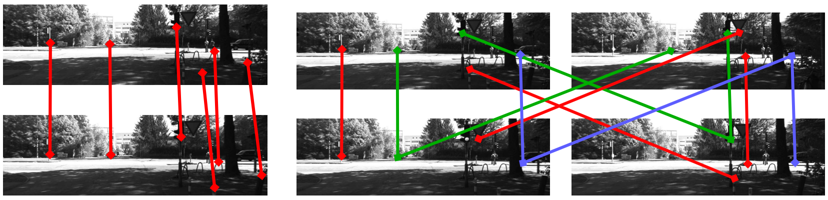
Chapter 2

Multiset-RANSAC

2.1 Introduction

RANSAC [1] is the most widely used method to eliminate outlier correspondences when a minimal solution is available. This method assumes that we have a single set of correspondences from which we can extract the correct model by iteratively generating candidate solutions from a predefined number of random correspondences and evaluating them according to some consensus metric. Depending on the flavor of RANSAC the consensus metric is built on different assumptions about the error distribution of inliers and outliers. In the most basic RANSAC version [1], the consensus of a candidate solution is evaluated by simply counting the number of inliers according to a predefined error threshold. MLESAC [27] computes the likelihood of candidate models, assuming that the error of inliers follows a Gaussian distribution and the error of outliers follows a uniform distribution over a specified range. An exhaustive range of different RANSAC approaches can be found in [28] that not only use different consensus metrics but also different sampling methods.

In many practical scenarios, the multi-sensor problems we are going to address deviate from the standard assumptions of RANSAC. In order to illustrate this difference, consider two problems: the relative pose between two views of a single camera (Fig. 2.1(a)), and the relative pose between two views of a stereo camera (Fig. 2.1(b)). The first problem can be solved with the 5-point algorithm [7]. We start by establishing a set of point correspondences between the two views, and then RANSAC iteratively samples 5 correspondences from this set. In the second problem there are many different types of point correspondences that can be established. Firstly, there can be correspondences across 2, 3, or 4 views. Addition-



(a) Single camera

(b) Stereo camera

Figure 2.1: (a) Two views of a single camera have only one type of point correspondences (red). (b) Two views of a stereo camera have many types of correspondences: 2-views (red), 3-views (green), 4-views (blue). Additionally 2-view and 3-view correspondences can be established between different sets of cameras. Images are taken from the KITTI Vision Benchmark Suite [30]

ally, correspondences between 2 and 3 views can be established between different combinations of cameras. Therefore, minimal algorithms for this problem include different types of correspondences, including: 6 pairwise correspondences from 2 different view pairs [29]; 3 triple correspondences from different views [17]; three 2-view and one 4-view correspondences [19]; and a mix of 4-view, 3-view, and 2-view correspondences [20]. In these cases, sampling with RANSAC must be done in a more structured way.

These differences in the sampling procedure change the underlying assumptions of traditional RANSAC methods. For the stereo relative pose problem, this issue was briefly addressed in [19]. The authors show that the number of iterations must be computed differently when different datasets have different inlier ratios, and that 4-view and 2-view outliers/inliers must be weighted differently when computing an overall consensus metric.

In this chapter we extend the observations in [19] to different sampling scenarios and to different consensus metrics, namely maximum likelihood (MLESAC [27]) and maximum a posteriori estimation (MAPSAC [31]).

2.2 Problem Formulation

Assume a generic problem in which we want to find the model T that best fits into a dataset $\mathcal{D}$ containing K samples, some of which are inliers while the others are outliers. A standard approach from the RANSAC family iteratively selects a subset $\mathcal{S}$ with s random samples from $\mathcal{D}$ and uses a model generator $\mathsf{T}_C = g(\mathcal{S})$ to find the model that best fits $\mathcal{D}$, according to some consensus metric.

Now consider a new scenario where we have N datasets $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_N\}$. Assume also that two steps of random sampling are required to generate a candidate model $\mathsf{T}_C = g(\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_M)$. First we select M datasets from $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_N\}$, with $M \leq N$, and then from each of them we select subsets $\{\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{D}_M\}$ with $s_1, s_2, ..., s_M$ samples respectively, such that the total number of selected samples is $s = \sum_{i=1}^{M} s_i$. Throughout this thesis we refer to this problem as multiset-RANSAC.

There are some sampling scenarios in multiset-RANSAC that require special attention. In some cases we cannot arbitrarily select M random datasets $\mathcal{D}_i$ due to the nature of the problem. This is the case in the relative pose between stereo cameras from 3 point correspondences, where we must sample a 2-view, a 3-view, and a 4-view correspondence [19]. In this thesis we focus on multiset-RANSAC problems that are not of this nature, i. e., problems where any arbitrary combination of M datasets $\mathcal{D}_i$ can be used to generate a candidate model.

Additionally, when the number of datasets $\mathcal{D}_i$ is the same as the number of datasets required for the model generator, i. e. $M = N$, the first sampling step is just a random matching between all datasets $\mathcal{D}_i$ and the amounts of samples s_i that will be extracted from each of them. This case is also more prone to some over-fitting problems that will be discussed later.

In the remaining sections of this chapter we discuss the necessary adaptations to RANSAC, MLESAC, and MAPSAC when dealing with multiple datasets, which we designate by multiset-RANSAC, multiset-MLESAC, and multiset-MAPSAC respectively.

2.3 Multiset-RANSAC

In the original version of RANSAC it is assumed that inlier samples have a uniform error distribution over some bounded interval. All samples with an error greater than a threshold t is considered an outlier. In this case the evaluation cost of each candidate model is simply the total number of outliers (Fig. 2.2(a)). For the multiset-RANSAC approach we can use the same evaluation metric by summing up the outliers in all datasets. We might also consider tuning different thresholds for each dataset, when it makes sense in a particular problem.

In standard RANSAC the number of iterations n is determined by guaranteeing that at least one model was generated from only inlier samples with a probability p, set to a value close to 1. The sampling process is approximated by a succession of s Bernoulli trials, i. e., sampling with replacement, and n is updated in each iteration by

$$n = \frac{\log(1-p)}{\log(1-\gamma^s)} \tag{2.1}$$

where s is the number of samples and γ is the probability of randomly selecting one inlier sample, i. e., the inlier ratio of the dataset. This value is updated in each iteration according to the best available model.

In the multiset-RANSAC case n must be computed differently since the sampling process is different and each dataset $\mathcal{D}_i$ might have a different inlier ratio γ_i. The probability of obtaining an inlier by first selecting a random dataset $\mathcal{D}_i$ and then selecting a random sample from it is

$$p_{in} = \frac{1}{N}\sum_{k=1}^{N}\gamma_k \tag{2.2}$$

If we approximate the second sampling process by a succession of Bernoulli trials, the probability of selecting s_i inliers by first selecting a random dataset $\mathcal{D}_i$ and then selecting s_i random samples from it is

$$p_{ins} = \frac{1}{N}\sum_{k=1}^{N}\gamma_k^{s_i} \tag{2.3}$$

If we further approximate the first sampling process by a succession of Bernoulli trials with replacement, the complete multiset-RANSAC sampling process is simplified to selecting a random dataset M times and for each of them successively selecting s_1, s_2, ..., s_M samples. The probability of selecting only inliers in this process is

$$p_{inall} = \prod_{j=1}^{M}\frac{1}{N}\sum_{k=1}^{N}\gamma_k^{s_j} \tag{2.4}$$

We want to guarantee that after n multiset-RANSAC iterations the probability of never selecting only inliers is smaller or equal to a very small probability $1-p$, i. e.

$$\left(1-p_{inall}\right)^n \leq 1-p. \tag{2.5}$$

Therefore, the minimum number of RANSAC iterations must be

$$n = \frac{\log\left(1-p\right)}{\log\left(1-p_{inall}\right)}. \tag{2.6}$$

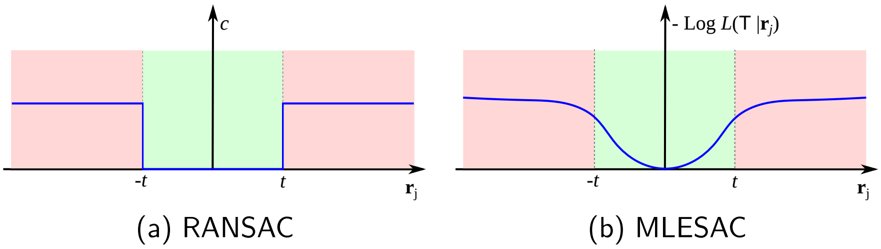


Figure 2.2: Cost evaluation metrics for a model T, given a sample d_j with residue r_j. The threshold t separates inliers (green) from outliers (red). In RANSAC the cost function c only dependends on whether d_j is an inlier or not. In MLESAC the cost function approximates a least squares problem if $\mathbf{d}_j$ is an inlier and a has a constant maximum cost if $\mathbf{d}_j$ is an outlier.

Note however that when M is close to N the first sampling step cannot be approximated by a succession of Bernoulli trials. Specifically in the case that $M = N$ the above equation might grossly underestimate the number of required iterations. In this case, there are $N!$ possible dataset selections, and to obtain p_{inall} we must weight in the probability of each of them selecting only inliers. We revisit this problem in Chapter 5 for a particular scenario where $N = M = 2$.

All the results derived for computing the number of multiset-RANSAC iterations also extend to the multiset-MLESAC and multiset-MAPSAC formulations presented in the following sections.

2.4 Multiset-MLESAC

MLESAC [27] aims at finding the model T with minimum negative log-likelihood, given a set of measurements $\mathcal{D}$. Each sample $\mathbf{d}_j$ in $\mathcal{D}$ can be put into one of two subsets: the inliers $\mathcal{I}$ or the outliers $\mathcal{O}$.

The residue of samples in $\mathcal{I}$ follows a Gaussian distribution $N(0, \sigma)$. A model T, given an inlier sample $\mathbf{d}_j$ with residue $r_j^{\mathcal{I}}$, has a likelihood

$$L(\mathsf{T}|r_j^{\mathcal{I}}) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-|r_j|^2}{2\sigma^2}} \tag{2.7}$$

The samples from $\mathcal{O}$ are observations independent from the model, and their residue follows a uniform distribution over an interval $[-\frac{v}{2}, \frac{v}{2}]$. A model T, given an outlier sample $\mathbf{d}_j$ with residue $r_j^{\mathcal{O}}$ has a constant likelihood

$$L(\mathsf{T}|r_j^{\mathcal{O}}) = \frac{1}{v} \tag{2.8}$$

Each sample from dataset $\mathcal{D}$ follows a mixed distribution of inliers and outliers (Fig. 2.2(b)) and therefore the likelihood $L(\mathsf{T}|r_j)$ of a model T, given a random sample $\mathbf{d}_j$ from $\mathcal{D}$ with residue r_j is

$$L(\mathsf{T}|r_j) = \left(\gamma \left(\frac{1}{\sqrt{2\pi}\sigma}\right) e^{\frac{-|r_j|^2}{2\sigma^2}} + \frac{1-\gamma}{v}\right) \tag{2.9}$$

where γ is the probability of d_j being an inlier, i. e., the inlier ratio in $\mathcal{D}$.

The MLESAC problem can now be formulated by considering the negative log-likelihood of T given all K samples in $\mathcal{D}$

$$\min_{\mathsf{T}} - \sum_{j=1}^{K} \log L(\mathsf{T}|r_j) \tag{2.10}$$

Note that the inlier ratio γ is updated in each iteration using expectation maximization using the following constraint:

$$\gamma_i = \frac{1}{K} \sum_{i=1}^{K} Pr(r_j^{\mathcal{I}}|\gamma) \tag{2.11}$$

where γ is initialized to 0.5 on the left side of the equation and is iteratively updated until convergence.

We now consider the multiset-MLESAC problem. When sampling from N different datasets we aim at maximizing the likelihood of model T given datasets $\mathcal{D}_1$, ..., $\mathcal{D}_N$, each of them with a number of samples K_i, an inlier standard deviation σ_i, an outlier range v_i, and an inlier ratio γ_i. In this case the likelihood of a model T, given a sample $\mathbf{d}_{i,j}$ from dataset $\mathcal{D}_i$ and with a residue $r_{i,j}$ is

$$L(\mathsf{T}|r_{i,j}) \left(\gamma_i \left(\frac{1}{\sqrt{2\pi}\sigma_i}\right) e^{\frac{-|r_{i,j}|^2}{2\sigma_i^2}} + \frac{1-\gamma_i}{v_i}\right) \tag{2.12}$$

The multiset-MLESAC problem for N datasets can now be formulated as

$$\min_{\mathsf{T}} - \sum_{i=1}^{N} \sum_{j=1}^{K_i} \log L(\mathsf{T}|r_{i,j}) \tag{2.13}$$

Note that to compute γ_1, ..., γ_N in each iteration we have to solve N expectation maximization problems with the form of equation 2.11.

After an accurate model has been found, the inliers can be selected by checking for each sample if

$$\gamma_i L(\mathsf{T}|r_{i,j}^{\mathcal{I}}) > (1-\gamma_i)L(\mathsf{T}|r_{i,j}^{\mathcal{O}}) \tag{2.14}$$

which can be rewritten as

$$|r_{i,j}|^2 < -2\sigma_i^2 \ln \frac{\sqrt{2\pi}\sigma_i(1-\gamma_i)}{\gamma_i v_i} \tag{2.15}$$

The most notable difference when we step from a standard MLESAC formulation to multiset-MLESAC is that different datasets might have different inlier ratios γ_i. This reflects a practical scenario in which some datasets are consistently more reliable than others. Multiset-MLESAC is able to capture those differences by estimating separate values γ_i for each dataset, which in turn results in a different cost function and inlier threshold for each dataset.

2.5 Multiset-MAPSAC

The MLESAC formulation can be further generalized to a maximum a posteriori problem (MAPSAC [31]). While [31] does a very exhaustive bayesian analysis of random sampling for geometric problems, we are only interested in its key observation that an algorithm from the RANSAC family does not only estimate the parameters of model T but also an additional set of latent parameters, namely deciding whether each sample is an inlier or an outlier through the expectation maximization of the inlier ratio γ. Taking this into account we formulate the multiset-MAPSAC problem as

$$\max_{\mathsf{T},\gamma_1,...,\gamma_N} Pr(\mathsf{T},\gamma_1,...,\gamma_N|\mathbf{R}_1,...,\mathbf{R}_N) \tag{2.16}$$

where $\mathbf{R}_1, ..., \mathbf{R}_N$ represent the residues of all samples from $\mathcal{D}_1, ..., \mathcal{D}_N$ respectively, which follow the mixed inlier-outlier distribution described in the previous section. This formulation can be re-written as

$$\max_{\mathsf{T},\gamma_1,...,\gamma_N} Pr(\gamma_1,...,\gamma_N,\mathsf{T})Pr(\mathbf{R}_1,...,\mathbf{R}_N|\mathsf{T},\gamma_1,...,\gamma_i) \tag{2.17}$$

Note that although this is a MAP formulation, it is not a step-by-step generalization of the MAPSAC method described in [31], which deals with the marginalization of parameter γ and the effect of additional latent parameters, e.g. reconstructed 3D points. In this thesis we do not take these issues into account.

When compared to multiset-MLESAC, equation 2.17 has an additional prior on the model and the latent parameters $Pr(\gamma_1, ..., \gamma_N, \mathsf{T})$. Prior knowledge about the model T is a very specific issue in each application scenario and we ignore it in the context of multiset-MAPSAC. Our main motivation behind this formulation is to account for prior knowledge about the inlier ratios γ_i. While multiset-MLESAC assumes that parameters γ_i are independent from each other, with multiset-MAPSAC we want to account for the possibility that this is not the case. In Chapter 5 we encounter this issue and detail a particular set of scenarios in which using prior knowledge on the inlier ratios is essential to obtain accurate results.

2.6 Multiset-RANSAC/MLESAC/MAPSAC in Practice

The application of this framework to the particular problems in this thesis is discussed case by case in each of the corresponding Chapters:

LRF-Camera Calibration (Chapter 3): We use standard RANSAC for validating our algorithm, however, we explain how multiset-RANSAC can be used to improve results in future work.

US-Camera Calibration (Chapter 4): We use multiset-RANSAC in all experimental validations.

Camera Networks (Chapter 5): This chapter presents the most extensive discussion and validation of multiset-RANSAC. This is a challenging case where the multiset-MAPSAC formulation is often required.

Chapter 3

Extrinsic calibration of a camera and a laser-range finder

3.1 Introduction

There are many systems and applications that combine perspective cameras and invisible 2D Laser-Rangefinder (LRF). A non-exhaustive list of examples includes the acquisition of ground-based city models by using an LRF for obtaining structure, and a camera for rendering texture [32]; the fusion of laser shape features with visual appearance for object classification [33] and pedestrian detection [34]; or the joint use of camera and laser for recognition and modeling of landmarks in outdoor self-localization and mapping [35]. In all these cases the fusion of the two sensor modalities requires knowing in advance the relative pose between camera and laser for projecting the depth readings into the images. Our article addresses this extrinsic calibration problem.

The number of published works in the extrinsic calibration of a camera and a LRF is relatively small. The most broadly used method was proposed by Zhang and Pless in [36], and describes a practical procedure where a checkerboard pattern is freely moved in front of the two sensors as shown in Fig. 3.1. The poses of the checkerboard are computed from plane-to-image homographies [37], and the camera coordinates of the planes are related with laser depth readings for establishing a set of linear constraints in the extrinsic calibration parameters. The solution of the system of equations provides an initial estimate for the relative rotation and translation, that is subsequently refined by iterative minimization of the re-projection error (similar to bundle adjustment [38]). Zhang's algorithm suffers from two major drawbacks:

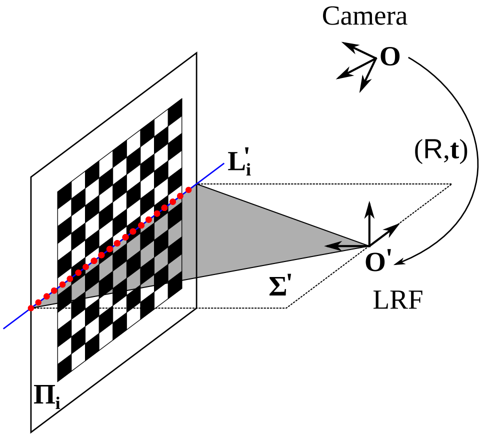


Figure 3.1: Input data for estimating the relative pose between camera and LRF reference frames with origins in $\mathbf{O}$ and $\mathbf{O}'$. The checkerboard plane is represented in camera coordinates $\mathbf{\Pi}_i$ that are determined from the plane-image homography. The LRF reads the depth of the points lying on the line where the pattern intersects the scan plane $\mathbf{\Sigma}'$. The calibration problem is formulated as the registration of planes $\mathbf{\Pi}_i$ and co-planar lines $\mathbf{L}'_i$ that are fitted to the depth readings.

(i) the system of linear equations does not directly enforce the rotation matrix to be in *SO(3)*, which often leads to poor initialization that cause the iterative estimation to run into local minima; and (ii) the closed-form algorithm requires at least 5 input planes being clearly a non-minimal solution for the calibration problem.

This paper proposes a minimal solution for the described extrinsic calibration that estimates the rigid displacement between camera and LRF from 3 input planes. We fit lines to the laser depth readings and carry the Euclidean registration of 3 planes with 3 co-planar lines in an optimal and closed-form manner. Our main contribution is this new registration algorithm that is used as an efficient hypothesis generator in a RANSAC paradigm [1] for robust camera-LRF calibration. The minimal solution is tested in simulation and its singularities are discussed. Experiments using both synthetic and real data show that the proposed calibration method outperforms the state-of-the-art [36] in terms of robustness, accuracy, and required number of input planes.

3.1.1 Related Work

This article is closely related to Zhang's work [36], where the extrinsic camera-LRF calibration is achieved by freely moving a checkerboard pattern. The procedure is simple to execute, and the checkerboard images can be used in parallel for calibrating the camera intrinsics [39, 40]. Like in [36], it is possible to use our method to jointly refine the intrinsic and extrinsic calibration during the final global optimization step. However, we do not discuss this feature and assume that the camera intrinsics are accurately known at all time. The calibration method presented in [41] is conceptually equivalent to [36], as is acknowledged by its authors.

Alternatives to Zhang's method can only be applied to a limited set of situations. Some contributions assume specific setups, like the LRF mounted on a calibrated rotating platform [42], or prior information, like an initial pose obtained through physical measurements [43,44]. In other cases additional inertial data is used [45,46]. A minimal solution for the extrinsic calibration of a camera and a LIght Detection And Ranging sensor (LIDAR) has been recently proposed [47]. The method uses a planar pattern for establishing correspondences between points in the LIDAR and lines in the image. The calibration problem is formulated as the 3D registration of co-planar points with planes intersecting into a single point. This leads to a system of polynomial equations that is solved using Macaulay resultants, obtained from 6 input images. Another minimal solution was proposed to calibrate both intrinsic and extrinsic parameters of a camera and a visible range finder [48]. In this case it is easy to make data associations between laser depth readings and visible laser dots projected on the camera without using a calibration target. This procedure requires the LRF-camera system to acquire 3 dot associations in 5 different positions. These last two approaches cannot be directly extended to the invisible LRF because they also use additional sensory information for the data association.

Since we formulate the camera-LRF calibration as the problem of aligning planes with co-planar lines, the article also relates with the literature in 3D Euclidean registration and related topics. In particular, we use previous results in registering two clouds of 3 or more 3D points [13]; in estimating the camera pose from the images of 3 or more 3D points (the so called Perspective-n-Pose (*PnP*) problem) [4,49]; and in determining from 3 correspondences the relative rotation between two views with known baseline [50]. Olsson et al. have recently proposed in [51] a Branch-and-Bound framework to solve different Euclidean registration problems: point-to-point, point-to-line, and point-to-plane. Within this topic, the recent work of Ramalingam et al. [52] in minimal solutions for the registration of points and planes is specially relevant. It is possible to adapt their algorithm for aligning 3 planes with 3 generic lines where each line is parameterized as a pair of points. Although such approach can eventually lead to a minimal solution for camera-LRF calibration, we propose an alternative registration algorithm that simplifies the problem by conveniently exploring the fact that the lines are co-planar.

3.2 Notation

In addition to the notation presented in Chapter 1, the remaining sections from this Chapter use additional notation. We use a prime symbol to designate geometric entities represented in the LRF reference frame, e.g. $\mathbf{\Sigma}'$, $\mathbf{L}'_i$. Plain letters are used to designate geometric entities in the camera reference frame, e.g. $\mathbf{\Pi}_i$, $\mathbf{d}_i$. The same letter with and without prime, e.g. $\mathbf{\Sigma}$, $\mathbf{\Sigma}'$ represent the **same** geometric entity under the camera and LRF reference frames, respectively.

3.3 The Calibration Problem

Consider a camera and a LRF for which the local coordinate systems have origin in $\mathbf{O}$ and $\mathbf{O}'$ as shown in Fig. 3.1. The extrinsic calibration aims at determining the rigid transformation T such that:

$$\begin{pmatrix} \mathbf{Q}' \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \mathsf{R} & \mathbf{t} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{pmatrix}}_{\mathsf{T}} \begin{pmatrix} \mathbf{Q} \\ 1 \end{pmatrix}, \tag{3.1}$$

where $\mathbf{Q}$ and $\mathbf{Q}'$ are respectively non-homogeneous point coordinates in camera and LRF reference frames, R denotes a rotation matrix, and $\mathbf{t}$ is the translation vector.

In [36] the calibration is carried from N images of a checkerboard pattern that is freely moved in front of the two sensors. Let $\mathbf{\Pi}_i$ be the homogeneous representation of the calibration plane in camera coordinates, that is estimated from plane-to-image point correspondences [37]:

$$\mathbf{\Pi}_i \sim \begin{pmatrix} \mathbf{n}_i \\ 1 \end{pmatrix}, \qquad i = 1, 2 \ldots N \tag{3.2}$$

For each plane $\mathbf{\Pi}_i$, the LRF provides depth readings of a set of 3D points $\mathbf{Q}'_{ij}$ that lie on the line where the checkerboard intersects the scan plane $\mathbf{\Sigma}'$. Let the non-homogeneous coordinates in the laser reference frame be

$$\mathbf{Q}'_{ik} \; = \; \begin{pmatrix} x_{ik} \\ y_{ik} \\ z_{ik} \end{pmatrix} , \qquad k = 1, 2 \ldots K_i$$

Zhang and Pless assume, without loss of generality, that $\mathbf{\Sigma}'$ is coincident with the Y plane. By inverting equation 3.1 and taking into account that y_{ik} is always zero, it follows that:

$$\mathbf{Q}_{ik} = \underbrace{\mathsf{R}^{\mathsf{T}} \begin{pmatrix} 1 & 0 & \\ 0 & 0 & -\mathbf{t} \\ 0 & 1 & \end{pmatrix}}_{\mathsf{H}} \mathring{\mathbf{Q}}_{ik} \, , \tag{3.3}$$

with $\mathbf{Q}_{ik}$ being the point representation in camera coordinates and

$$\mathring{\mathbf{Q}}_{ik} \; = \; \begin{pmatrix} x_{ik} \\ z_{ik} \\ 1 \end{pmatrix} .$$

Since the points detected by the laser are in the checkerboard pattern, then the following must hold

$$\mathbf{\Pi}_i{}^{\mathsf{T}} \, \mathbf{Q}_{ik} \; = \; 0 \, .$$

Replacing by the results of equations 3.2 and 3.3, it yields that

$$\mathbf{n}_i^{\mathsf{T}} \, \mathsf{H} \, \mathring{\mathbf{Q}}_{ik} \; = \; -1 \, , \quad \forall_{i,k} \tag{3.4}$$

Note that H is a metric homography, and therefore it has a fixed scale such that the magnitude of the translation $\mathbf{t}$ is consistent with the depth measurements of the LRF.

In summary, the checkerboard planes $\mathbf{\Pi}_i$, expressed in camera coordinates, and the points $\mathbf{Q}'_{ik}$, represented in LRF coordinates, define a set of linear constraints in the entries of matrix H that encodes the rigid displacement between the two sensors. In [36], Zhang and Pless propose to compute H in a DLT-like manner [53], and factorize the result into the rotation R and translation $\mathbf{t}$. Unfortunately the linear estimation of matrix H is carried without enforcing the structure of equation 3.3. This means that in general the direct factorization does not provide a valid rotation matrix R, and a non-optimal projection into *SO(3)* is required. Moreover, for each calibration plane $\mathbf{\Pi}_i$ there are only two constraints of the form of equation 3.4 that are linearly independent. Since matrix H has 9 entries, then the estimation requires $N \geq 5$ calibration planes. The fact that the calibration algorithm is non-minimal and sub-optimal often leads to erroneous results as shown later.

We propose instead to fit lines to the laser points and formulate the problem as the 3D registration of a set of co-planar lines $\mathbf{L}'_i$ with a set of planes $\mathbf{\Pi}_i$. In other words, we aim to find the rotation R and the translation $\mathbf{t}$ such that the planes $\mathbf{\Pi}'_i$, given by

$$\mathbf{\Pi}'_i = \underbrace{\begin{pmatrix} \mathsf{R} & \mathbf{0} \\ -\mathbf{t}^\mathsf{T}\mathsf{R} & 1 \end{pmatrix}}_{\mathsf{T}^{-\mathsf{T}}} \mathbf{\Pi}_i \,, \qquad i = 1, 2 \dots N \tag{3.5}$$

go through the lines $\mathbf{L}'_i$.

In [24] we show that, by formulating the problem in the dual 3D space, the rotation and translation can be estimated separately. Finding the rotation is equivalent to determining the relative orientation between two views with known baseline. As proved in [50], this last problem can be cast as a standard *P3P* problem [4] admitting at most 8 distinct solutions. For each rotation solution there is a corresponding translation that can be found by solving an additional system of linear equations. In this thesis we use a more intuitive and simple approach to show the equivalence between finding the rotation and the *P3P* problem.

3.4 P3P Problem

Consider the Figure 3.4 that represents a calibrated pinhole camera observing three 3D points $\mathbf{P}'_{12}$, $\mathbf{P}'_{13}$, $\mathbf{P}'_{23}$ as 2D image projections with homogeneous coordinates $\mathbf{d}_{12}$, $\mathbf{d}_{13}$, $\mathbf{d}_{23}$. Consider as well that the coordinates of the 3 points are known in a world reference frame while the pose of the camera is unknown. The *P3P* problem consists in finding the 3 depths α_{12}, α_{13}, α_{23} between the principal point of the camera and the 3 points $\mathbf{P}'_{12}$, $\mathbf{P}'_{13}$, $\mathbf{P}'_{23}$.

The unknown depths can be obtained by solving the following quadratic system of equations

$$\begin{cases} \alpha_{12}^2 + \alpha_{13}^2 - \alpha_{12}\alpha_{12}\mathbf{d}_{12}{}^\mathsf{T}\mathbf{d}_{13} = ||\mathbf{P}'_{12} - \mathbf{P}'_{13}||^2 \\ \alpha_{12}^2 + \alpha_{23}^2 - \alpha_{12}\alpha_{12}\mathbf{d}_{12}{}^\mathsf{T}\mathbf{d}_{23} = ||\mathbf{P}'_{12} - \mathbf{P}'_{23}||^2 \\ \alpha_{13}^2 + \alpha_{23}^2 - \alpha_{12}\alpha_{12}\mathbf{d}_{13}{}^\mathsf{T}\mathbf{d}_{23} = ||\mathbf{P}'_{13} - \mathbf{P}'_{23}||^2 \end{cases} \quad . \tag{3.6}$$

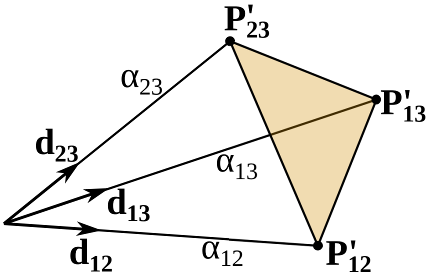


Figure 3.2: The *P3P* problem: determining depths α_{12}, α_{13}, α_{23} given the 3D points $\mathbf{P}'_{12}$, $\mathbf{P}'_{13}$, $\mathbf{P}'_{23}$ and the image ray directions $\mathbf{d}_{12}$, $\mathbf{d}_{13}$, $\mathbf{d}_{23}$ of a calibrated camera at an unknown pose.

This system has in up to 8 possible solutions, of which only up to 4 have positive depth values. The solutions containing negative depths are normally ignored since they are physically invalid in the context of pinhole camera projection. Different solutions have been described in the literature to solve this system [4,54,55]. In this thesis we use Grunert's method [56] described in [4].

After determining the depths α_{12}, α_{13}, α_{23} we can obtain the coordinates of the 3D points $\mathbf{P}_{ij} = \alpha_{ij}\mathbf{d}_{ij}$ in the reference frame of the camera and use 3D point registration to find the pose of the camera in the world reference frame (absolute orientation problem [13]).

3.5 Minimal Solution

Consider Fig. 3.3(a), that represents 3 line measurements $\mathbf{L}'_1$, $\mathbf{L}'_2$, $\mathbf{L}'_3$ in the LRF cutting plane Σ'. In general, each pair of lines $\mathbf{L}'_i$, $\mathbf{L}'_j$ intersects at a point $\mathbf{P}'_{ij}$. Now consider the 3 planes $\mathbf{\Pi}'_1$, $\mathbf{\Pi}'_2$, $\mathbf{\Pi}'_3$ that represent the calibration target in O'. Each plane $\mathbf{\Pi}'_i$ contains its correspondent line $\mathbf{L}'_i$. In general, each pair of planes $\mathbf{\Pi}'_i$, $\mathbf{\Pi}'_j$ intersects at a line $\mathbf{l}'_{ij}$ that goes through the correspondent point $\mathbf{P}'_{ij}$ (Fig. 3.3(b)). The 3 planes also intersect at a single point $\mathbf{m}'$. The lines $\mathbf{l}_{ij}$ (i. e. the representation of lines $\mathbf{l}'_{ij}$ in O) can be easily determined by intersecting the known planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$.

Consider now a virtual pinhole camera with reference frame V (Fig. 3.3(b)) such that its orientation is the same as the real camera at O and its center of projection is point $\mathbf{m}'$. Consider also that for each line $\mathbf{l}_{ij}$ its direction vector, denoted by $\mathbf{d}_{ij}$, represents an image ray in the virtual camera of the 3D point $\mathbf{P}'_{ij}$ with depth α_{ij}. It is now evident that determining all three depths α_{ij} is a problem with the same

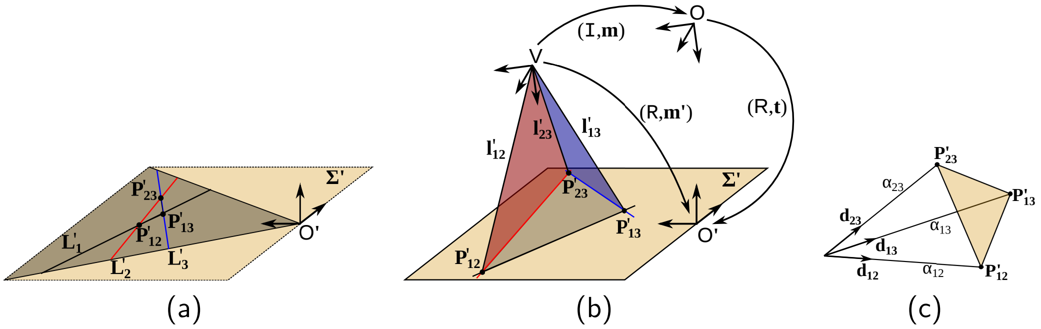


Figure 3.3: From registration of planes and co-planar lines to the *p3p* problem: (a) Each pair of lines $\mathbf{L}'_i$, $\mathbf{L}'_j$ measured by the LRF intersects at point $\mathbf{P}'_{ij}$; (b) Each pair of planes $\mathbf{\Pi}'_i$, $\mathbf{\Pi}'_j$ intersects at line $\mathbf{l}_{ij}$; the three planes intersect at point $\mathbf{m}'$; we define a virtual camera V with projection center at $\mathbf{m}'$ and the same orientation as the real camera at O; (c) the classic *P3P* problem is obtained by defining each direction $\mathbf{d}_{ij}$ of line $\mathbf{l}_{ij}$ as an image point projection of $\mathbf{P}'_{ij}$ in the virtual camera V.

geometric structure as the *P3P* problem with image points $\mathbf{d}_{12}$, $\mathbf{d}_{13}$, $\mathbf{d}_{23}$ and 3D points $\mathbf{P}'_{12}$, $\mathbf{P}'_{13}$, $\mathbf{P}'_{23}$ (Fig. 3.3(c)).

This system can be solved by standard *P3P* algorithms [4]. Unlike in the original problem however, the scalar unknowns α_{ij} do not have the physical meaning of depth and are allowed to take negative values. Hence there can be up to 8 distinct solutions to equation 3.6.

After determining the depths α_{ij} we can compute the rotation R and translation $\mathbf{m}'$ from V to O' by solving the absolute orientation problem [13]. Once the pose of the virtual camera is known the transformation T from the real camera to the LRF can be obtained as shown in Fig. 3.3(b) by

$$\mathsf{T} = \begin{pmatrix} \mathsf{R} & \mathbf{m}' \\ 0 & \mathbf{1} \end{pmatrix} \begin{pmatrix} \mathtt{I} & -\mathbf{m} \\ 0 & \mathbf{1} \end{pmatrix} = \begin{pmatrix} \mathsf{R} & \mathbf{m}' - \mathsf{R}\mathbf{m} \\ 0 & \mathbf{1} \end{pmatrix} \tag{3.7}$$

3.5.1 Degenerate configurations

This problem has degenerate configurations that arise in two situations. There is a translation ambiguity when two lines $\mathbf{l}'_{ij}$, $\mathbf{l}'_{ik}$ are parallel, placing the point $\mathbf{m}'$ at infinity. A camera center at infinity is a well known degeneracy of the *P3P* problem (Fig. 3.4(a)). The *danger cylinder* configuration is another known degeneracy of the *P3P* problem [4]. This happens when the point $\mathbf{m}'$ belongs to the cylinder

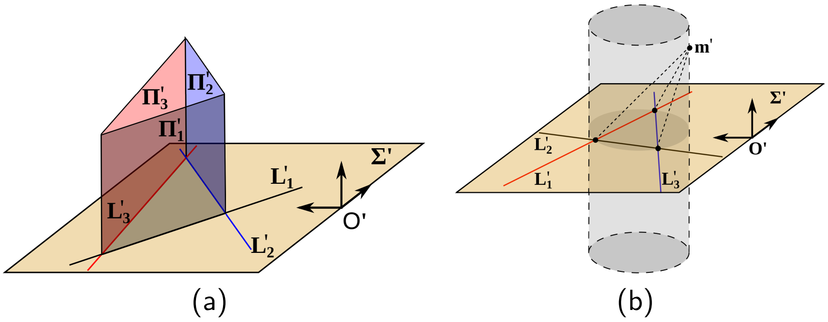


Figure 3.4: **Degenerate configurations:** (a) the lines where the checkerboard planes intersect are parallel and $\mathbf{m}'$ is at the infinity; (b) the intersection point $\mathbf{m}'$ of the 3 checkerboard planes lies in the *danger cylinder* [4] defined by the intersection points of lines $\mathbf{L}'_1$, $\mathbf{L}'_2$, $\mathbf{L}'_3$.

that contains points $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$ and is orthogonal to the plane defined by them (Fig. 3.4(b)).

3.5.2 Outline of the registration algorithm

This algorithm determines the relative pose T that aligns a set of planes $\mathbf{\Pi}_i$ with a set of co-planar lines $\mathbf{L}'_i$. Its inputs are 3 correspondences $(\mathbf{\Pi}_i, \mathbf{L}'_i)$ and its output is a set of up to 8 relative poses T defined by their rotations R and translations $\mathbf{t}$.

1. For each two planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$ in the camera reference frame determine the direction $\mathbf{d}_{ij}$ of their intersecting line $\mathbf{l}_i$, by cross-multiplying the plane normals.

2. For each two lines $\mathbf{L}'_i$, $\mathbf{L}'_j$ in the laser reference frame determine their point of intersection $\mathbf{P}'_{ij}$.

3. Formulate the *P3P* problem with each $\mathbf{P}'_{ij}$ as a 3D point, and each direction $\mathbf{d}_{ij}$ as its respective image point.

4. Solve the *P3P* problem using any standard approach [4] to obtain the rotation R and the translation $\mathbf{m}'$. There are up to 8 solutions.

5. Determine the point intersection $\mathbf{m}$ of planes $\mathbf{\Pi}_i$

6. Determine the translation $\mathbf{t}$ as $\mathbf{m}' - \mathsf{R}\mathbf{m}$.

3.6 Extrinsic Calibration Algorithm

Section 3.5 derives a closed-form algorithm for computing the $M \leq 8$ rigid transformations that align 3 planes with 3 co-planar lines. We now show how this new registration method can be used for obtaining the extrinsic calibration between the camera and the laser.

Let's recall that the inputs for calibration are the planes $\mathbf{\Pi}_i$ with $i = 1, 2 \ldots N$, expressed in camera coordinates, and the points $\mathbf{Q}'_{ik}$ with $k = 1, 2 \ldots K_i$, represented in non-homogeneous LRF coordinates. The application of plane-line registration requires fitting lines $\mathbf{L}'_i$ to the points $\mathbf{Q}'_{ik}$ using a standard regression method. If the number of input planes is $N = 3$, then the registration algorithm provides $M \leq 8$ solutions $\mathsf{T}^{(m)}$ with $m = 1, 2 \ldots M$, but we cannot decide about the rigid displacement between the two sensors without further information. For the case of $N > 3$, each triplet of plane-line correspondences gives rise to a set of solutions, and the correct relative pose T can be found using an hypothesize-and-test framework as detailed in section 3.6.1. Note that in practice accurate calibrations are obtained with a relatively low N (between 5 and 12) and therefore it is feasible to test all combinations of 3 correspondences.

In both situations the final calibration estimates can be further refined by minimizing the re-projection errors in the camera and LRF using iterative non-linear optimization. Thisfinal refinement step is discussed in section 3.6.3.

3.6.1 Initial Estimation

Consider $N > 3$ correspondences between planes $\mathbf{\Pi}_i$ and lines $\mathbf{L}'_i$. The initial estimate T for the extrinsic calibration is obtained as follows:

1. Initialize the solution $\mathsf{T} = \mathtt{I}$
2. Select 3 correspondences and apply the algorithm of section 3.5.2 for finding the transformations $\mathsf{T}^{(m)}$ that align lines and planes ($m = 1, 2 \ldots M$).
3. For each solution $\mathsf{T}^{(m)}$, compute the LRF coordinates $\mathbf{\Pi}'^{(m)}_j$ of the remaining $N - 3$ planes, and determine the Euclidean distance $d_j^{(m)}$ in the dual space between $\mathbf{\Pi}'^{(m)}_j$ and the corresponding line $\mathbf{L}'_j$.
4. Rank each solution $\mathsf{T}^{(m)}$ by assigning the score

$$score(\mathsf{T}^{(m)}) = \sum_j \max(t, d_j^{(m)}) \,,$$

where t is a pre-defined threshold. This operator is similar to the one used in the MSAC robust estimator proposed in [27].

5. If $score(\mathsf{T}) > score(\mathsf{T}^{(m)})$, then make $\mathsf{T} = \mathsf{T}^{(m)}$ and consider as inliers the correspondences for which $d_j^{(m)} < t$ (the 3 correspondences that generated the solution have $d_j^{(m)} = 0$).
6. Until all combinations of 3 correspondences are tested, return to step 2 for a new iteration.

Since the number of input correspondences is usually small ($N < 20$), we run an exhaustive search where all possible plane-line triplets are considered as solution generators. For a large N the hypothesize-and-test can be performed in a Random Sample Consensus manner in order to keep the computation tractable [1].

3.6.2 Multiset-RANSAC

In the previous section we formulated this calibration problem in terms of a single dataset containing N plane-line correspondences, each of them corresponding to one LRF-Camera acquisition. Although in most cases this is sufficient to accurately solve the calibration problem, there is still room for improvement by formulating the problem without explicitly computing the LRF lines $\mathbf{L}'_i$ before using a hypothesize-and-test framework.

Instead we can use a multiset-RANSAC framework if we formulate the calibration problem as N datasets, each of them containing K_i points $\mathbf{Q}'_{ik}$ and a plane $\mathbf{\Pi}_i$. In this case an initial estimation can be obtained by iteratively generating hypotheses with a two step sampling procedure: first select 3 LRF-Camera acquisitions and then select two points from each dataset. Lines $\mathbf{L}'_i$ are estimated independently in each iteration.

This formulation can be useful for two main reasons. The estimation of LRF lines $\mathbf{L}'_i$ can improve slightly, since the best generated hypothesis will use the lines that best fit to all acquisitions instead of using independent line estimations. But most interestingly, we can use multiset-RANSAC to decrease the necessity of human input during the calibration procedure by performing automatic line detection from the LRF depth measurements. In practice the LRF acquisitions output many undesired outlier measurements outside the extent of the calibration plane. These outliers are usually removed manually, since the background might contain other lines (e.g. walls) that would mislead an automatic line detector. With multiset-RANSAC we

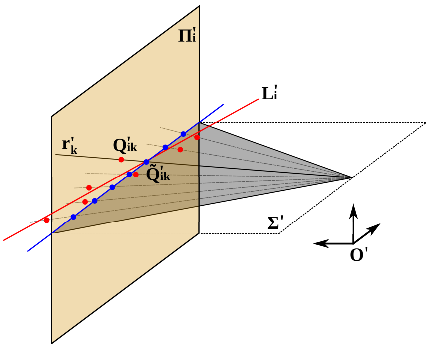


Figure 3.5: The LRF reconstructs points $\mathbf{Q}'_{ik}$ (red) by measuring depth along the radial directions $\mathbf{r}'_k$. The minimization is carried over the distance between points $\mathbf{Q}'_{ik}$ and points $\widetilde{\mathbf{Q}}'_{ik}$ (blue). The latter are obtained by mapping the inlier planes $\mathbf{\Pi}_i$ into the LRF reference frame and intersecting the result with the radial directions $\mathbf{r}'_k$.

can skip this manual selection step and allow the generated hypotheses to choose the correct measurements of the calibration plane.

We do not pursue this approach further in this thesis, and thus it serves as a reference for future research directions. However, in the next Chapter we use multiset-RANSAC to perform automatic line detection in a similar calibration problem.

3.6.3 Iterative least-squares refinement

The initialization procedure provides an extrinsic calibration T and a set of plane-line correspondences that are classified as inliers. The calibration accuracy can be further improved by minimizing the re-projection errors in the camera and/or LRF using iterative least squares refinement similar to bundle adjustment [38].

The LRF measures depth along a set of radial directions $\mathbf{r}'_k$ that are uniformly distributed in the scan plane $\mathbf{\Sigma}'$ around the projection center. Please note that we assume that this projection center is not coincident with the origin $\mathbf{O}'$ of the laser reference frame. The depth readings enable to reconstruct the points $\mathbf{Q'}_{ik}$ that give rise to the lines $\mathbf{L}'_i$ that are considered in the 3D registration. After obtaining an initial calibration estimate T, each inlier plane $\mathbf{\Pi}_i$ is mapped into the LRF reference frame using equation 3.5, and the resulting plane $\mathbf{\Pi}'_i$, expressed in laser coordinates, is intersected with the radial lines $\mathbf{r}'_k$ yielding a set of points $\widetilde{\mathbf{Q}}'_{ik}$ (see Fig. 3.5). The LRF residue to be minimized is the sum of the square distances between the points

$\widetilde{\mathbf{Q}}'_{ik}$ and the points $\mathbf{Q}'_{ik}$ that were originally reconstructed from the depth readings:

$$e_{LRF} = \sum_i \sum_k ||\mathbf{Q}'_{ik} - \widetilde{\mathbf{Q}}'_{ik}||^2$$

The errors in the orientation of planes $\mathbf{\Pi}_i$ can also be minimized with the standard cost function used in camera calibration methods [39], i. e. the re-projection error of the checkerboard grid corners $\mathbf{X}_{il}$ against their 2D point image detections $\mathbf{h}_{il}$

$$e_{CAM} = \sum_i \sum_l ||\mathbf{h}_{il} - \mathsf{K}\mathsf{H}_i\mathbf{X}_{il}||^2 \tag{3.8}$$

where K is the camera intrinsics matrix and H_i is the metric homography from plane $\mathbf{\Pi}_i$ to the camera such that

$$\mathsf{H}_i = \begin{pmatrix} \mathbf{r}_{1,i} & \mathbf{r}_{2,i} & \mathbf{t}_i \end{pmatrix} \tag{3.9}$$

$$r_{3,i} = \mathbf{r}_{1,i} \times \mathbf{r}_{2,i} \tag{3.10}$$

$$\mathbf{\Pi}_i = \begin{pmatrix} \mathbf{r}_{3,i} \\ \mathbf{t}_i{}^{\mathsf{T}} r_{3,i} \end{pmatrix} . \tag{3.11}$$

The LRF and the camera residues can be jointly minimized with the following cost function

$$\min_{\mathsf{T},\mathbf{r}_{3,1},\mathbf{t}_1,\ldots,\mathbf{r}_{3,N},\mathbf{t}_N} e \;=\; e_{LRF} + \kappa\, e_{CAM} \;, \tag{3.12}$$

where κ is a weighting parameter that should be adjusted to normalize the variance of camera and LRF residue distributions. The minimization is carried with respect to the extrinsic calibration T and the pose of the inlier planes $\mathbf{\Pi}_i$ expressed in camera coordinates. The camera residue e_{CAM} depends both on the planes $\mathbf{\Pi}_i$ and the camera intrinsics K. We will assume that K is accurately known but, like in [36], this formulation can be potentially used to refine simultaneously the intrinsic and the extrinsic calibration, by considering the independent parameters of K (focal length, skew, aspect ratio and principal point) as variables to be refined.

3.7 Experiments with Synthetic Data

A first set of experiments is conducted in a simulation environment that considers a 0.25° resolution LRF and a 1280×960 resolution pin-hole camera. The LRF is assumed to be stationary and the pin-hole camera is randomly placed in a pre-defined region according to a uniform distribution. The camera placement is such that there is always a significant overlap between the fields of view of the two sensors. Given the camera and the LRF, we simulate a set of N checkerboard planes with random poses. Once again the plane placement is such that guarantees intersection with the laser scan plane and a minimum number of grid points visible in the camera. We add Gaussian noise to both the image grid points and the laser depth readings. Please note that the pose of the checkerboard plane affects the number of points $\mathbf{Q'}_{ik}$ that are reconstructed by the LRF, and hence the accuracy of the lines $\mathbf{L}'_i$ used for the plane-line registration.

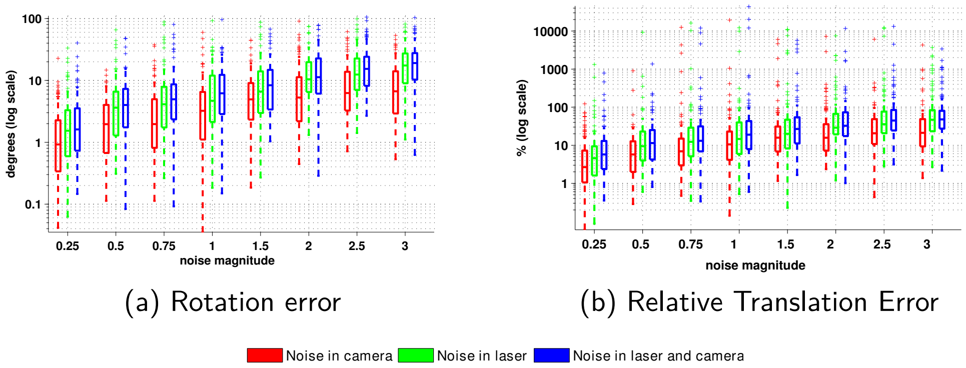
This simulation environment provides input data for performing the extrinsic calibration. The estimations for the relative rotation R and a translation $\mathbf{t}$ are compared with the ground-truth R_{GT} and $\mathbf{t}_{GT}$. The accuracy is typically quantified by the angular magnitude of the residual rotation $\mathsf{R}^\mathsf{T}\mathsf{R}_{GT}$, and by the relative translation error $||\mathbf{t} - \mathbf{t}_{GT}|| \,/\, ||\mathbf{t}_{GT}||$.

3.7.1 Extrinsic calibration with minimum data ($N = 3$)

In this experiment the extrinsic calibration is carried using $N = 3$ calibration planes. For each trial, we randomly generate one camera pose and three checkerboard planes. The simulated image points and laser depths are used as calibration input after adding Gaussian noise. Since the noise affects both the estimation of planes $\mathbf{\Pi}_i$ and lines $\mathbf{L}'_i$, the result of the plane-line registration is in general different from the correct rigid displacement between the two sensors [1]. Fig. 3.6 shows the distribution of these errors in 100 independent trials, for increasing amounts of noise in the camera and/or laser. For visualization purposes, we do not plot results without noise, however, we also simulated this situation to perform a sanity check, yelding to maximum errors of 0.0021% for translation and 0.0012 degrees for rotation. The median error is below 10^{-10}% for translation, and below double precision for rotation.

The figure shows compact error distributions with few outliers, which suggests that the calibration algorithm is numerically stable. For low noise levels we can still

[1] As stated in section 3.5, the plane-line registration has up to 8 analytical solutions and it is not possible to choose the one corresponding to the extrinsic calibration without further information. In the synthetic experiments we always consider the solution that is closest to the ground-truth



(a) Rotation error

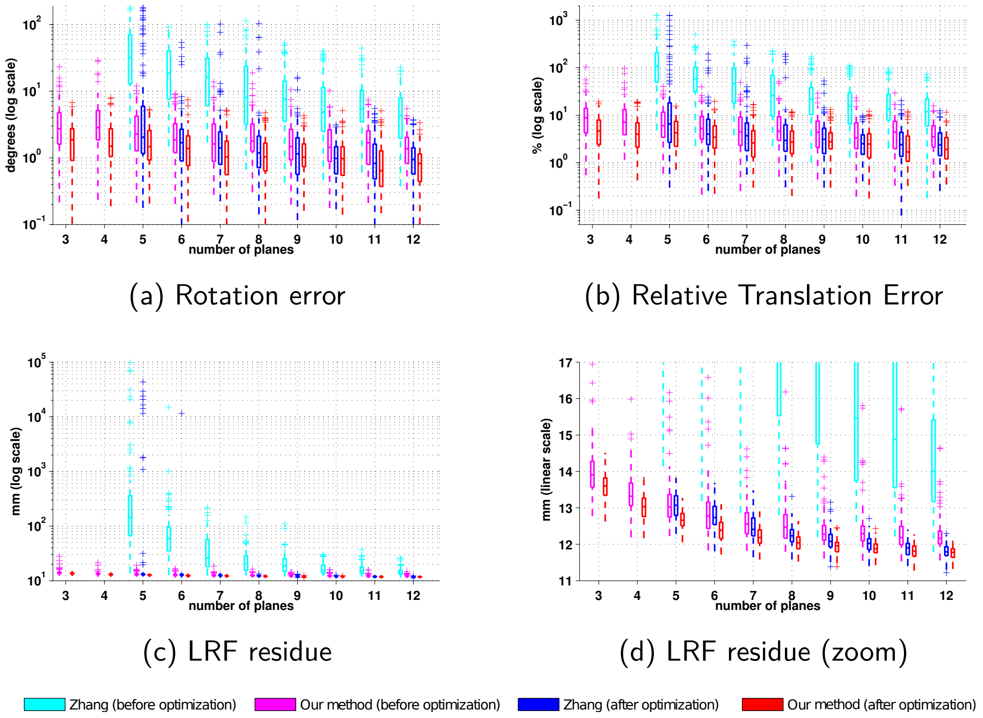
(b) Relative Translation Error

Figure 3.6: Error distribution when the extrinsic calibration is carried using $N = 3$ calibration planes (minimum solution). The labels in the horizontal axis refer to the standard deviation of the added Gaussian noise. We consider 1 pixel steps for the camera, and 15 mm steps for the LRF (e.g. a magnitude of 0.25 corresponds to image noise of 0.25 pixels and laser noise of 3.75 mm).

detect some outliers, that mainly result from the degenerate configuration depicted by situation 1 in figure 3.4. Input data close to this configuration occur due to generating input planes with bounded orientation variations, so that both sensors can fully detect them. The extrinsic calibration accuracy decreases with increasing amounts of noise, but this degradation is relatively smooth. The rotation estimation seems to be less sensitive to noise than the translation. This is partially explained by the fact that the rotation is computed first and its error propagates to the translation component. In overall terms the results are satisfactory and prove that, if the measurements are not too noisy and the checkerboard orientations are carefully chosen, then the extrinsic camera-LRF calibration can be achieved in practice using only $N = 3$ input planes.

3.7.2 Comparison with Zhang's method

The simulation framework is now used to compare our algorithm against the calibration method proposed in [36]. For the sake of fairness, Zhang's method is implemented in a hypothesize-and-test framework that is in every respect similar to the one described in section 3.6.1, except that it requires 5 plane-line correspondences to generate a hypothesis. The experiment considers a variable number N of calibration planes and, for each case, it runs 100 calibration trials and compares the error distributions. The noise is constant and set to 1 pixel in the camera and 15 mm in



(a) Rotation error

(b) Relative Translation Error

(c) LRF residue

(d) LRF residue (zoom)

Figure 3.7: Calibration using synthetic data. We compare our algorithm against the method presented in [36] when the number N of calibration planes increases. The additive Gaussian noise has constant standard deviation of 1 pixel in the camera and 15 mm in the LRF.

the LRF. Fig. 3.7 shows the results for the two methods before and after running the non-linear optimization discussed in section 5.6.3.

A careful analysis of the graphics show that our algorithm provides much better initial estimates both in terms of the extrinsic calibration error and residue in the laser. This fact significantly decreases the chances of divergence during the iterative optimization step, specially when the number N of calibration planes is small. It can be observed that, for the case of $N = 5$ and $N = 6$, while our calibration results never diverge significantly from the ground-truth, the final calibration obtained using Zhang's method is often completely erroneous. The stability and final accuracy of the two methods tends to become similar for a large number of input planes ($N > 8$).

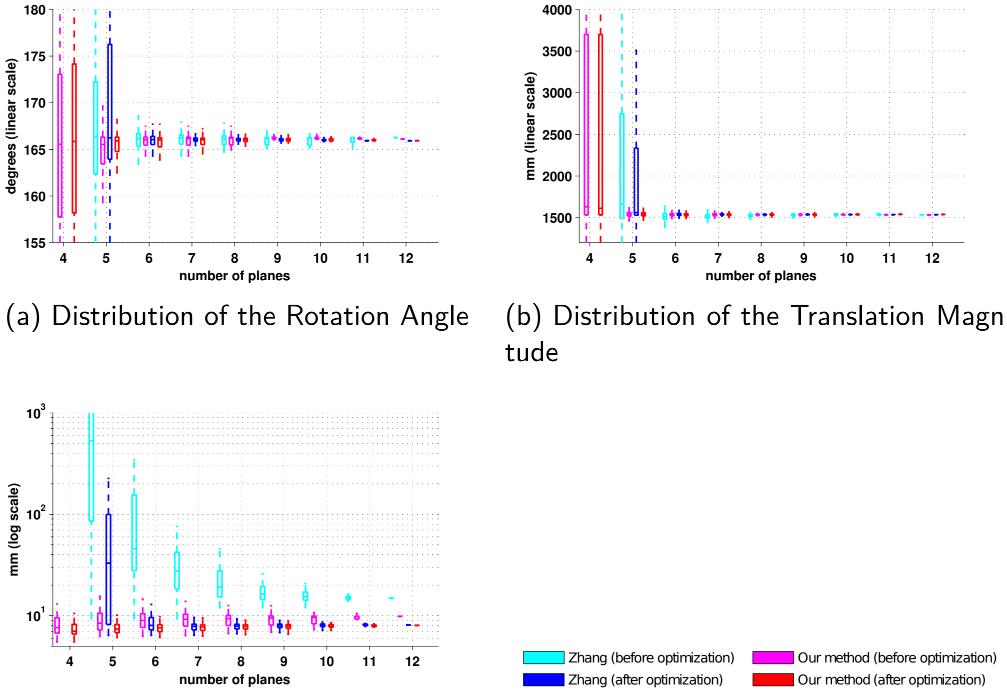
3.8 Experiments with Real Data

In this experiment we set up a SICK LMS 200 [57] and a camera at fixed positions, and acquire 12 calibration frames by moving a checkerboard pattern in front of the two sensors. It should be taken into account that acquiring frames with a wide range of plane orientations and distances to the LRF generally yelds more accurate calibration results. The camera intrinsic parameters and the homogeneous coordinates $\mathbf{\Pi}_i$ of the planes are estimated using the intrinsic calibration software described in [40]. Since the re-projection error in the plane-to-image homographies is typically very low (0.2 pixels), there is no advantage in considering the planes $\mathbf{\Pi}_i$ in the final iterative refinement described in section 5.6.3. Thus, we decided to optimize the cost function of equation 3.12 only with respect to the LRF residue ($\kappa = 0$). Like in the previous experiment, the extrinsic calibration is carried for an increasing number of calibration planes using both our method and the hypothesize-and-test version of Zhang's algorithm. We consider for each N all possible combinations of the 12 frames, which means that the number of trials is

$$\#_N = \frac{12!}{N!\,(12-N)!}$$

In the absence of reliable ground-truth, Fig. 3.8 shows the distribution of the achieved calibration results. More specifically Fig. 3.8(a) refers to the angle of the rotation R, Fig. 3.8(b) concerns the magnitude of the translation vector $\mathbf{t}$, and Fig. 3.8(c) depicts the LRF residues.

From Fig. 3.8 it comes that our minimal solution outperforms Zhang's approach when the number of input planes is small. While the former requires 4-to-5 planes for providing accurate estimation results, the latter needs 7 or more planes for achieving a reliable calibration. Fig. 3.9 confirms that, for the case of $N = 5$, Zhang's extrinsic calibration is in general non plausible. This is justified by the noise in the input data and the existence of frames with few laser readings that are unable to fully constrain the estimation problem. As the number of input planes increases, the hypothesize-and-test procedure discards these frames as outliers, and the output of the two methods converges to the same result. Nevertheless, it is important to remark that this only happens after the final refinement step. While Zhang's initialization is often a coarse estimate of the correct rigid displacement, our closed-form solution is always very close to the global optimum, and the improvements in accuracy achieved by the iterative refinement are somewhat marginal. In addition our approach, being a minimal solution, requires the testing of less hypothesis which



(a) Distribution of the Rotation Angle

(b) Distribution of the Translation Magnitude

(c) LRF residue

Figure 3.8: Calibration using real data. The graphics show the distributions of the calibration results obtained using our method and Zhang's algorithm [36]

is an indisputable advantage in terms of computational complexity. For the case of $N = 12$, an exhaustive search of the solution space requires 220 trials, while the same search with Zhang's method corresponds to 792 tests.

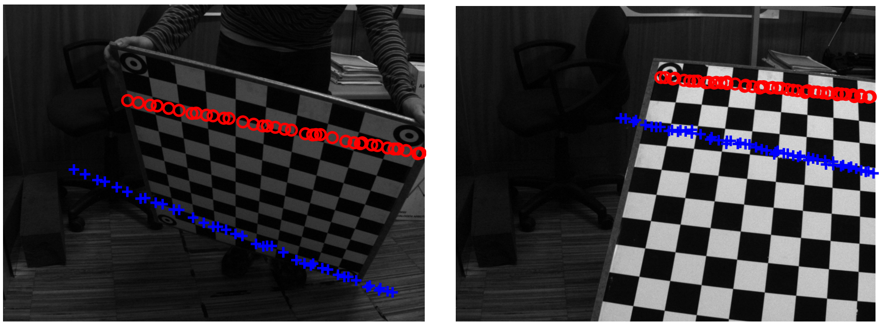


Figure 3.9: Projection of the LRF points into the images using the extrinsic calibration results obtained from 5 sampled planes with our method (circles) and Zhang's algorithm (crosses).

Chapter 4

Extrinsic Calibration between a Camera and an Ultra-sound array

4.1 Introduction

A 2D ultrasound array (US) is a sensor that measures the sound impedance along a single measurement plane and displays it as a 2D image where different grayscale levels represent different materials with a different sound impedance (B-scan). The acoustic impedance for each pixel is obtained by emitting ultrasound pulses within a certain frequency range and measuring the magnitude and time-response of the echo signals. The frequency of the emitted pulses will affect the resolution and the measurement range of the US probe, i. e., high frequencies (8 – 20 MHz) are able to measure impedances with fine detail (high mm/pixel ratio) but are absorbed by tissue within a very short period, while low frequencies (2 – 4 MHz) measure impedances with a lower resolution but are able to propagate through longer distances. High frequencies are therefore used for close-range imaging (e.g. musculoskeletal) while low frequencies are used to image deeper regions within the body (e.g. cardiac, obstetrics).

There is an increasing number of medical applications that require 6D pose tracking of an US. These applications include registration with 3D models [58], assisted coordination with other medical instruments [59], and also 3D ultrasound reconstruction [60].

There are different alternatives available for tracking the pose of an US, namely magnetic sensors [61], optical trackers [62, 63], and robot manipulators [64]. All these alternatives have different degrees of applicability depending on the particular scenario [65], however, all of them require to rigidly attach the US to a tracking device and solve a hand-eye calibration problem to determine the transformation that maps US image coordinates to the 3D Euclidean reference frame of the tracking device.

This hand-eye calibration procedure usually involves using a known calibration target that is measured by the US under different poses until there are enough geometric constraints to determine a unique hand-eye transformation. The calibration target can be a set of thin wires [66–68], a surface with known shape [69], a robot-manipulated instrument [64], or a single plane surface [70]. An extensive review of different calibration methods can be found in [65]. Recent contributions predominantly use wire-based targets, particularly z-shaped wires that allow for a greater flexibility in capturing US measurements under different poses [67, 68].

In this chapter we focus on the plane-based calibration introduced in [70]. In this method the US measures line-sections of a single plane under different poses, and the calibration problem is formulated as the 3D Euclidean registration of co-planar lines in the US image with planes in an external reference frame. This method has the significant advantage of not requiring to build a complex calibration target with precise measures. A plane can be easily obtained, e.g., as the bottom of a water filled tank. Despite these advantages, this method is rarely used in practice due to two major drawbacks: 1) Measuring planes at moderately skewed angles produces undesired US surface reflections that make line detections less accurate 2) All available methods are based on iterative local optimization, which means that we either collect a huge number of measurements to avoid local minima or use other means to obtain a sufficiently accurate initialization.

The first problem can be reduced by using a Cambridge phantom [70] or by measuring a thin membrane instead of a thick tank wall [71]. We will address the second problem by proposing a closed-form minimal solution to the hand-eye calibration from 4 plane measurements. This minimal algorithm can be used as a model generator within multiset-RANSAC to produce a robust calibration that is suitable for initializing a local optimization method. To the best of our knowledge, this is the first closed-form solution to the plane-based US calibration.

In our formulation we assume that US tracking is performed with a stationary camera and a visual marker attached to the US. In previous formulations [70] the pose of the calibration plane is unknown in the reference frame of the tracking device, and thus it must be estimated simultaneously with the hand-eye parameters. The camera-based tracking makes it easy to know the plane position beforehand, reducing the complexity of the calibration problem. Our formulation is also valid for other tracking devices, provided that the pose of the calibration plane can be previously determined.

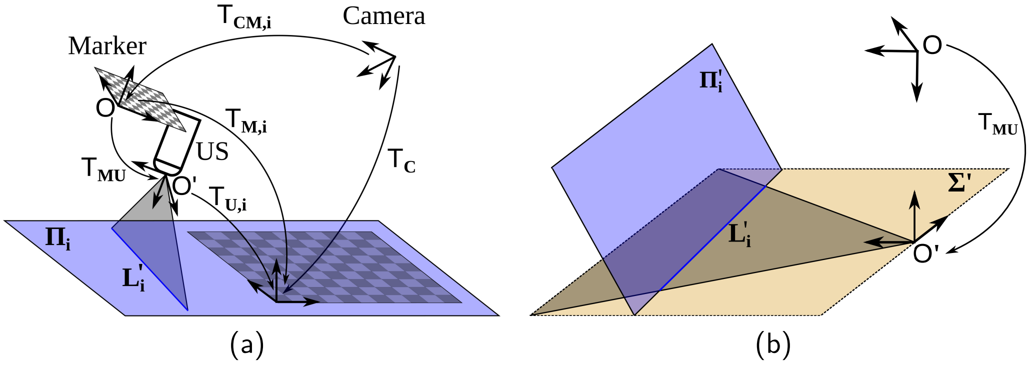
4.2 Notation

In addition to the notation presented in Chapter 1, the remaining sections from this Chapter use additional notation. We use a lower-case letter with prime symbol to designate geometric 2D entities represented in the US image, e.g. $\mathbf{x}'_i$, $\mathbf{p}'_i$, and an upper-case letter with prime symbol for 3D entities represented in the US reference frame, e.g. $\mathbf{\Sigma}'$, $\mathbf{L}'_i$. An asterisk denotes 3D geometric entities in the camera reference frame, e.g. $\mathbf{\Pi}^*_i$, $\mathbf{D}^*_i$. Plain letters are used to designate geometric entities in the visual marker reference frame, e.g. $\mathbf{\Pi}_i$, $\mathbf{D}_i$. The same letter with/without these additional symbols, e.g. $\mathbf{\Sigma}^*$, $\mathbf{\Sigma}'$, $\mathbf{\Sigma}$ represent the **same** geometric entity under the camera, the US, and the visual marker reference frames respectively.

4.3 Problem formulation

Consider a US sensor rigidly attached to a visual marker, whose local coordinate systems have origin at O' and O respectively, as shown in Fig. 4.1(a). The pose of the visual marker is detected by a camera whose reference frame has origin at O^*.

The calibration aims at mapping 2D homogeneous image coordinates $\mathbf{x}'_j$ in the US to 3D points $\mathbf{X}_j$ in the visual marker reference frame. This requires to determine both the intrinsic parameters K and the extrinsic parameters $\mathsf{T}_{\mathbf{UM}}$. One possible approach to do this would be to first determine the intrinsic parameters K using a calibration phantom with known dimensions and then perform extrinsic calibration using plane-line correspondences in a similar way to the LRF-camera problem of the previous chapter. In this thesis, however, we focus on the joint calibration of both intrinsic and extrinsic parameters as this allows to use a single calibration phantom and greatly simplifies the calibration procedure. A simple and fast work-flow during calibration is a key element in calibration of medical instruments since they often need to be performed in the operating room in a very controlled environment with strict temporal and spatial constraints.



$\mathsf{T}_{\mathbf{CM,i}}$
Camera
Marker
O
$\mathsf{T}_{\mathbf{M,i}}$
US
$\mathsf{T}_{\mathbf{MU}}$
O'
$\mathsf{T}_{\mathbf{U,i}}$
$\mathsf{T}_{\mathbf{C}}$
$\mathbf{\Pi_i}$
$\mathbf{L_i'}$
(a)
O
$\mathbf{\Pi_i'}$
$\mathsf{T}_{\mathbf{MU}}$
$\mathbf{\Sigma'}$
$\mathbf{L_i'}$
O'
(b)

Figure 4.1: (a) Calibration set-up: The US is rigidly attached to a visual marker that is tracked by a stationary camera. The US measures lines $\mathbf{L}_i'$ from a calibration plane $\mathbf{\Pi}$, while the camera measures the transformation $\mathsf{T}_{\mathbf{CM,j}}$ and maps the calibration plane to the Marker reference frame. (b) The US-Camera calibration problem can be formulated as the registration between planes $\mathbf{\Pi}_i$ represented in O and coplanar lines $\mathbf{L}_i'$ represented in O'.

The US intrinsic parameters K map 2D points $\mathbf{x}_j'$ in pixels to 3D points $\mathbf{X}_j'$ in metric coordinates, represented in O'

$$\mathbf{X}_j' = \mathsf{K}\mathbf{x}_j' \tag{4.1}$$

with

$$\mathsf{K} = \begin{pmatrix} f_x^{-1} & 0 \\ 0 & f_y^{-1} \\ 0 & 0 \end{pmatrix}, \quad f_x > 0, f_y > 0 \tag{4.2}$$

Note that, without loss of generality, we consider that points $\mathbf{X}_j'$ lie in the plane $z = 0$.

The extrinsic parameters $\mathsf{T}_{\mathbf{UM}}$ consist of rigid transformation with a rotation R and translation $\mathbf{t}$ that map points $\mathbf{X}_j'$ from the US reference frame to points $\mathbf{X}_j$

$$\mathbf{X}_j = \begin{pmatrix} \mathsf{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathsf{K}\mathbf{x}_j' \\ 1 \end{pmatrix} \tag{4.3}$$

The full intrinsic and extrinsic transformation from image coordinates $\mathbf{x}_j'$ to 3D coordinates $\mathbf{X}_i$ can be represented by

$$\mathbf{X}_j = \begin{pmatrix} \mathsf{A} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}_j' \\ 1 \end{pmatrix} \tag{4.4}$$

with

$$\mathsf{A} = \begin{pmatrix}\mathbf{a}_1 & \mathbf{a}_2\end{pmatrix} = \begin{pmatrix}\frac{\mathbf{r}_1}{f_x} & \frac{\mathbf{r}_2}{f_y}\end{pmatrix} \tag{4.5}$$

where $\mathbf{a}_1$, $\mathbf{a}_2$ are the first and second columns of A, and $\mathbf{r}_1$, $\mathbf{r}_2$ are the first and second columns of R.

Note that both K and R can be easily extracted from A, thus determining the 9 parameters in $\mathbf{a}_1, \mathbf{a}_2, \mathbf{t}$ solves the calibration problem. Additionally, since R is an orthogonal matrix the following quadratic constraint must be verified

$$\mathbf{a}_1{}^{\mathsf{T}}\mathbf{a}_2 = 0 \tag{4.6}$$

Therefore this problem can be solved minimally with 8 additional constraints on $\mathbf{a}_1, \mathbf{a}_2, \mathbf{t}$.

4.3.1 Line-Plane Registration

In an analogous way to the LRF-Camera problem from Chapter 3, this calibration problem can be formulated as the registration between lines $\mathbf{L}'_i$ on the US cutting plane Σ' and planes $\mathbf{\Pi}_i$ represented in O (Fig. 4.1(b)). The measurement of planes $\mathbf{\Pi}_i$ requires a stationary camera that can detect both the fixed plane $\mathbf{\Pi}^*$ and the varying pose $\mathsf{T}_{\mathbf{CM},j}$ of the visual marker.

If the intrinsics K are known beforehand this problem is equivalent to the LRF-Camera calibration problem and it can be solved with 3 plane-line correspondences using the algorithm presented in the previous chapter. However, in order to solve for both intrinsic and extrinsic parameters, the lines $\mathbf{L}'_i$ are affected by the two unknown scaling factors f_x, f_y and additional plane-line correspondences are required to solve the problem. As it will be shown in this section, a minimal solution can be obtained from 4 correspondences.

When there are N plane-line correspondences, in general each of the $\frac{N!}{(N-2)!}$ pairs of lines $\mathbf{L}'_i$, $\mathbf{L}'_j$ intersects at a point $\mathbf{P}'_{ij}$. Additionally, each of the $\frac{N!}{(N-2)!}$ pairs of planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$ intersects at a line $\mathbf{B}_{ij}$ and each triplet of planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$, $\mathbf{\Pi}_k$ intersects at a point $\mathbf{M}_{ijk}$. Particularizing for 4 plane-line correspondences there are 6 points $\mathbf{P}'_{ij}$ (Fig. 4.2(a)), 6 lines with directions $\mathbf{D}_{ij}$, and 4 points $\mathbf{M}_{ijk}$ (Fig. 4.2(b), 4.2(c)).

In this case we can think in terms of 4 virtual cameras V_{ijk} with projection centers at $\mathbf{M}_{ijk}$ and the same orientation as the visual marker with origin at $\mathbf{O}$ (Fig. 4.2(d)). Each of these cameras measures 3D points $\mathbf{P}_i$, $\mathbf{P}_j$, $\mathbf{P}_k$ as their image projections $\mathbf{D}_i$, $\mathbf{D}_j$, $\mathbf{D}_k$. Finding the pose of these virtual cameras solves our calibration problem.

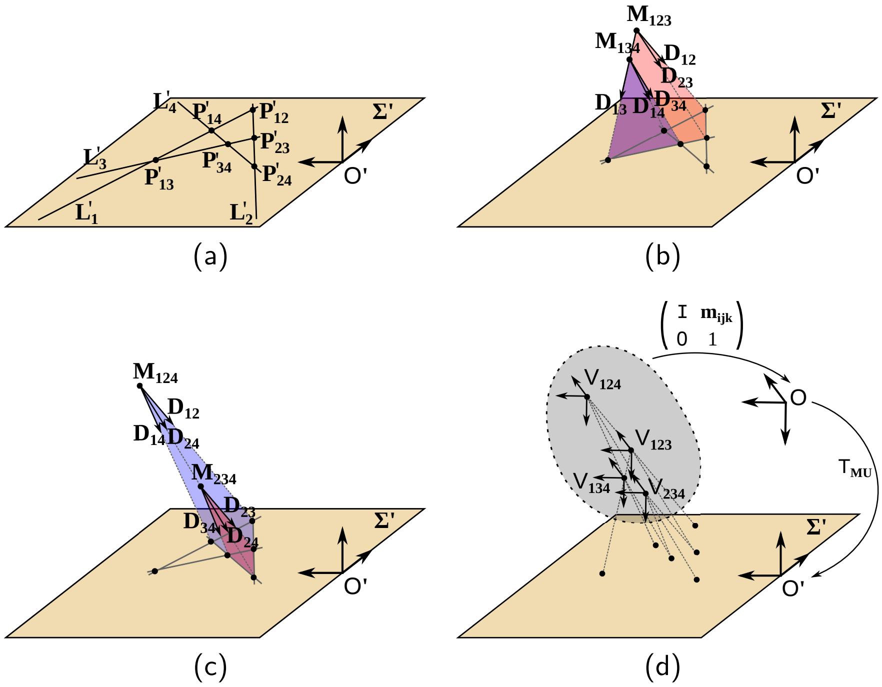


Figure 4.2: Registration of 4 Line-plane correspondences: (a) 4 lines $\mathbf{L}'_i$ intersect at 6 points $\mathbf{P}'_{ij}$; (b), (c) For each line $\mathbf{L}'_i$ there is a plane that intersects it. These 4 planes intersect at 6 lines with directions $\mathbf{D}_{ij}$ and 4 points $\mathbf{M}_{ijk}$; (d) Each point $\mathbf{M}_{ijk}$ can be thought as the principal point of a virtual pinhole camera V_{ijk} whose orientation is aligned with the reference frame at O. Each of these virtual cameras observes 3 points $\mathbf{P}_{ij}$ such that their image projection rays have directions $\mathbf{D}_{ij}$.

Unlike in the LRF-Camera Calibration, in this problem the location of 3D points $\mathbf{P}'_{ij}$ is not known due to the unknown US intrinsics K. Instead we can determine points $\mathbf{p}'_{ij}$ on the US image such that

$$\mathbf{P}'_{ij} = \mathsf{K}\mathbf{p}'_{ij} \tag{4.7}$$

For each virtual camera V_{ijk} the re-projection of a 3D point $\mathsf{K}\mathbf{p}'_{ij}$ is defined by the direction $\mathbf{D}_{ij}$ as

$$\mathbf{D}_{ab} \sim \begin{pmatrix}\mathsf{R} & \mathbf{t}-\mathbf{M}_{ijk}\end{pmatrix}\begin{pmatrix}\mathsf{K}\mathbf{p}'_{ab}\\ 1\end{pmatrix}, \quad a,b \in \{i,j,k\} \wedge a \neq b \tag{4.8}$$

This equation can be re-written as

$$\left(\mathbf{p}'_{ab}{}^{\mathsf{T}} \otimes [\mathbf{D}_{ab}]_\times \quad [\mathbf{D}_{ab}]_\times\right)\left(\mathbf{a}_1{}^{\mathsf{T}} \quad \mathbf{a}_2{}^{\mathsf{T}} \quad \mathbf{t}^{\mathsf{T}}\right)^{\mathsf{T}} = [\mathbf{D}_{ab}]_\times \mathbf{M}_{ijk} \tag{4.9}$$

Each instance of equation 4.9 puts 2 linear constraints on $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{t}$. There are only up to 8 linearly independent constraints of this form, while the calibration problem has 9 unknowns, thus equation 4.6 must be used to determine the remaining unknown parameter.

4.3.2 Outline of the minimal solution

We now summarize our minimal algorithm step by step. Consider as input 4 correspondences between planes $\mathbf{\Pi}_i$ in the visual marker reference frame and 4 lines $\mathbf{l}'_i$ in US image coordinates. The algorithm returns up to 2 solutions to the vectors $\mathbf{r}_1$, $\mathbf{r}_2$, $\mathbf{t}$ and the scalars f_x, f_y defined in equations 4.4 and 4.5.

1. Determine the line direction $\mathbf{D}_{ij}$ for the intersection of each pair of planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$ in the visual marker reference frame.

2. Determine the point of intersection $\mathbf{M}_{ijk}$ for each triplet of planes $\mathbf{\Pi}_i$, $\mathbf{\Pi}_j$, $\mathbf{\Pi}_k$ in the visual marker reference frame.

3. Determine the point of intersection $\mathbf{p}'_{ij}$ for each pair of lines $\mathbf{l}'_i$, $\mathbf{l}'_j$ in US image coordinates.

4. Find 8 linearly independent equations using instances of equation 4.9.

5. Solve the linear system up to one scalar unknown α, such that $\mathsf{A} = \alpha\mathsf{A}_1 + \mathsf{A}_2$ and $\mathbf{t} = \alpha\mathbf{t}_1 + \mathbf{t}_2$.

6. Determine α by solving equation 4.6. There are up to 2 solutions.

7. For each $(\mathbf{a}_1, \mathbf{a}_2, \mathbf{t})$ solution, find the positive values f_x, f_y such that $\mathbf{r}_1$,$\mathbf{r}_2$ from equation 4.5 have unitary norm.

4.3.3 Degenerate Configurations

From Fig. 4.2(d) we can observe that if we ignore the 2 unknown scale factors f_x, f_y our problem is equivalent to aligning a set of image rays from different projection centres with a set of co-planar 3D points. This is a particular case of the absolute orientation problem for generalized cameras [17]. The addition of the unknown scale

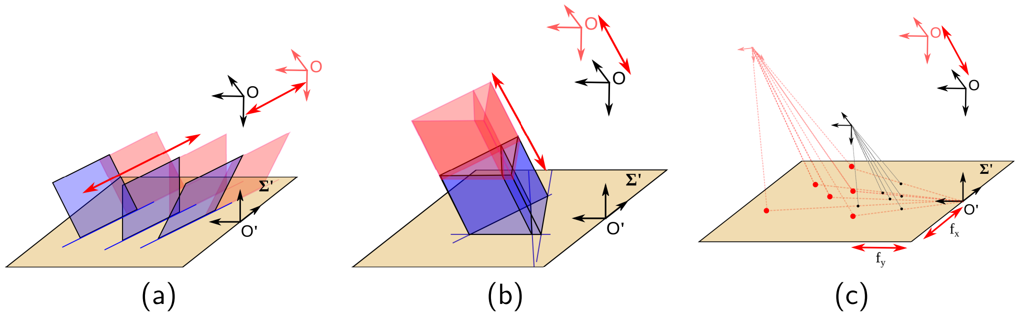
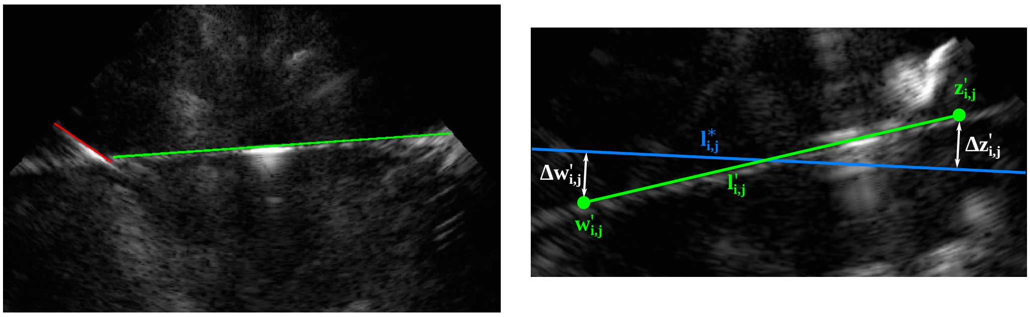


Figure 4.3: Degenerate configurations: (a) When the lines detected by the US are parallel there is an ambiguity in translation (b) When the planes measured by the visual tracker all intersect in parallel lines there is an ambiguity in translation (c) When the the planes measured by the visual tracker intersect at the same point there is an ambiguity in scale and translation

factors makes our problem similar in some aspects to the pose-and-scale problem for generalized cameras [72]. Although this problem is slightly different, the degenerate cases discussed in this paper are relevant in the context of our problem.

When all lines $\mathbf{L}'_i$ are parallel there is a translational ambiguity along the same direction (Fig. 4.3(a)). In our calibration problem we are in this configuration when all acquisitions are done by moving the US without rotation, or just with rotations that are either around the $x-$axis of the US reference frame or around an axis perpendicular to the calibration plane. When all the line directions $\mathbf{D}_{ij}$ are parallel, there is a translational ambiguity along the same direction (Fig. 4.3(b)). This happens in our problem when all planes $\mathbf{\Pi}_i$ have co-planar normals, i. e., all acquisitions are done by rotating the US along an axis parallel to the calibration plane. Finally, when all points $\mathbf{M}_{ijk}$ are coincident, there is a scale factor ambiguity (Fig. 4.3(c)). In this case we can always scale f_x,f_y in order to put the US further away or closer to the calibration plane. This case is analogous to the scale factor ambiguity that results from estimating motion/structure with a single camera [7,72]. This situation occurs when all acquisitions are done by rotating the US around a single point on the calibration plane.

To summarize, degenerate configurations can be avoided by measuring different regions of the calibration plane and by exploring all three degrees of freedom in rotation when moving the US between different acquisitions.



(a) (b)

Figure 4.4: Line detection on US images with Hough transform and PEARL: (a) The green line segment is a correct detection of the calibration plane while the red is an incorrect detection produced by an undesired artifact on the US image (b) Line segment detections are represented by start and end points $\mathbf{w}'_{i,j}$, $\mathbf{z}'_{i,j}$. multiset-RANSAC uses as error metric the orthogonal distances between these points and the re-projected line $\mathbf{l}^*_{i,j}$.

4.4 Practical considerations

4.4.1 Line Detection

We perform automatic line segment detection on US images with the Hough transform [73]. However, there are undesired reflections and artifacts that can clutter the measurements and produce Hough peaks for different lines. While we want to ignore spurious line detections, in many cases it is not guaranteed that the maximum Hough peak corresponds to the correct line. In order to deal with this problem we use the PEARL approach for multi-model fitting [74] to minimize the number of line detections, while still considering multiple line segment candidates in ambiguous cases (Fig. 4.4(a)). For each US image PEARL detects K_i line segments defined by their start and end points $\mathbf{w}'_{i,j}$, $\mathbf{z}'_{i,j}$ (Fig. 4.4(b)). We then can then use multiset-RANSAC to select the correct line candidates based on geometric constraints.

4.4.2 Multiset-RANSAC

multiset-RANSAC is used when there are more than 4 plane-line correspondences available. Consider N pairs of US and Camera image acquisitions. Each US acquisition contains K_i start and end points $\mathbf{w}'_{i,j}$, $\mathbf{z}'_{i,j}$ in homogeneous image coordinates that define a 2D line $\mathbf{l}'_{i,j}$. Each camera provides a single reliable estimation of the calibration plane $\mathbf{\Pi}_i$. The K_i samples from each US-Camera acquisition are represented by vectors $\mathbf{d}_{i,j} = \begin{pmatrix}\mathbf{w'}^{\mathsf{T}}_{i,j} & \mathbf{z'}^{\mathsf{T}}_{i,j} & \mathbf{\Pi}^{\mathsf{T}}_i\end{pmatrix}^{\mathsf{T}}$.

Candidate solutions are generated by first sampling 4 random US-Camera acquisitions and then from each of them sampling one random vector $\mathbf{d}_{i,j}$.

The residue metric $r_{i,j}$ for each sample $\mathbf{d}_{i,j}$, given a candidate solution $(\mathsf{K}, \mathsf{R}, \mathbf{t})$, is the squared sum of the orthogonal distances between points $\mathbf{w}'_{i,j}$, $\mathbf{z}'_{i,j}$ and the 2D line re-projection $\mathbf{l}^*_i$ of the calibration plane $\mathbf{\Pi}_i$ onto the image (Fig. 4.4(b))

$$r_{i,j} = ||\Delta\mathbf{z}'_{i,j}||^2 + ||\Delta\mathbf{w}'_{i,j}||^2 \tag{4.10}$$

4.4.3 Iterative least-squares refinement

A final refinement of the calibration solution is performed by iterative least-squares minimization using the Levenberg-Marquadt algorithm. After removing outliers with multiset-RANSAC, we convert each line segment into a discrete set of points $\mathbf{p}_{j,i}$ corresponding to each line pixel. We also re-project each plane $\mathbf{\Pi}$ onto the US image by representing it in the US reference frame and intersecting it with the plane $z = 0$. We minimize the total squared orthogonal distance between points $\mathbf{p}_{j,i}$ and the re-projected lines $\mathbf{l}^*_i$ in pixels (Fig. 4.4(b))

$$\min_{\mathsf{T},\mathsf{K}} \sum_i \sum_j \frac{\mathbf{p}_{j,i}{}^{\mathsf{T}}\mathbf{l}^*_i}{||\mathsf{D}\mathbf{l}^*_i||} \tag{4.11}$$

with

$$\mathsf{D} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{4.12}$$

4.5 Validation

We validate our calibration method in experiments with both synthetic and real data. In all experiments we use multiset-RANSAC and iterative least-squares refinement as detailed in the previous section.

For real data acquisition we use a TITAN SONOSITE ultrasound system with a 2–4 Mhz probe inside a water tank and a Grasshopper2 camera calibrated with the EasyCamCalib toolbox. We measure the bottom of the water tank directly, without using any additional method to increase the accuracy of US line detections, i. e. the Cambridge phantom [70] or the thin membrane method [71], and thus our results reflect a worst case scenario in terms of US measurement accuracy. In the synthetic experiment we try to approximate as close as possible to this set-up by defining virtual sensors with similar error magnitudes and measurement ranges.

We represent the US intrinsic parameters in terms of focal length $f = \sqrt{f_x f_y}$ and aspect ratio $a = \sqrt{\frac{f_x}{f_y}}$.

4.5.1 Synthetic data

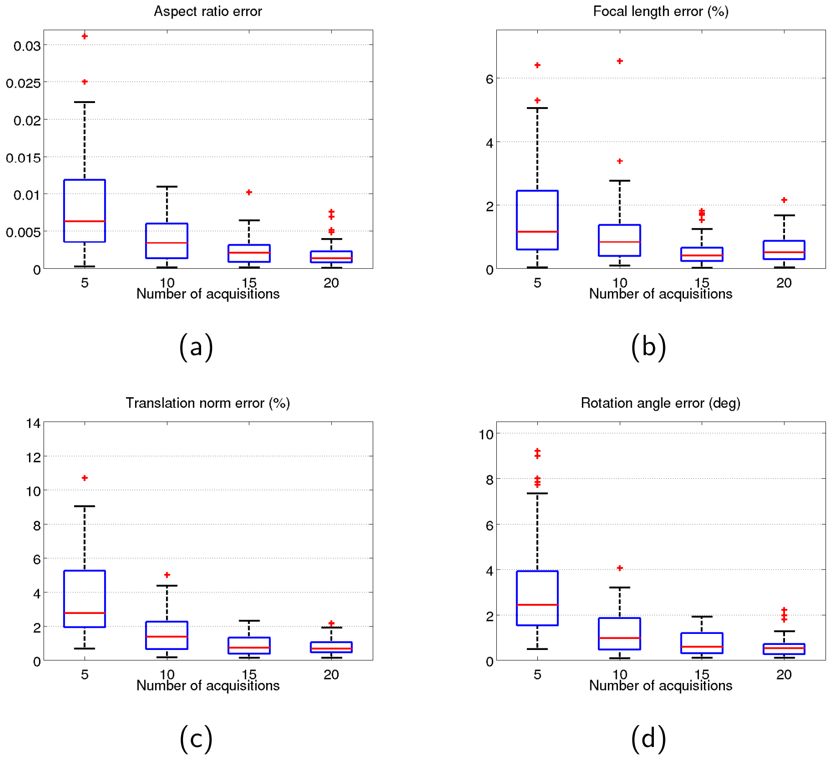
We use a simulated environment that contains a 1280×720 resolution pin-hole camera, a US with focal length $f_{GT} = 3$ and aspect ratio $a_{GT} = 1$, a visual marker attached to the US with 100 planar features, and a calibration plane at $z = 0$. The camera is placed at a fixed pose while the US is randomly placed in a pre-defined region such that the visual marker can always be seen by the camera. The US measurements range from distances between 80mm and 100mm to the calibration plane, and thus are representative of a $2--4$ MHz low frequecy probe. We simulate a set of N US/Camera measurements for different random US poses. We add Gaussian noise to both the visual tracker features measured by the camera ($\sigma = 0.2px$) and the US line measurements ($\sigma = 1px$). The noise in the camera is in line with the re-projection errors that can be achieved with the EasyCamCalib toolbox, while the noise in the US is set to a higher level due to the intrinsic difficulties in measuring plane cuts with an US [70]. Note that in this setup only one line is detected for each US measurement, and thus there are no outliers.

We perform 50 random calibration trials using different numbers of plane-line correspondences and compare the results against groundtruth values. Fig. 4.5 shows the error distribution for all calibration trials, quantified by the angular magnitude of the residual rotation $\mathsf{R}^\mathsf{T}\mathsf{R}_{GT}$, the relative translation error $||\mathbf{t} - \mathbf{t}_{GT}|| \,/\, ||\mathbf{t}_{GT}||$, the relative focal length error $\frac{|f_{GT}-f|}{f_{GT}}$, and aspect ratio error $|a - a_{GT}|$.

The results show that for the specified noise, even without outliers, using just 5 acquisitions is significantly inaccurate, with translation errors that can surpass 10%. Our calibration method converges to a stable solution for 15 images.

4.5.2 Real data

Our setup for real data acquisition consists of a stationary calibrated camera that tracks a checkerboard grid attached to the US. The US measurements are made inside a water filled tank. In a first step we place a checkerboard grid on the bottom of the empty tank to locate the calibration plane in the camera reference frame (Fig.



(a)

(b)

(c)

(d)

Figure 4.5: Error distributions with synthetic data. For each case we perform 50 trials and compare them against groundtruth values.

4.6(a)). Then we fill the tank with water and perform 20 synchronized US-Camera acquisitions with the US under different poses (Fig. 4.6(b)). Finally we validate our results with a 403 GS LE phantom (Fig. 4.6(c)).

In an analogous way to the synthetic set-up, we perform 50 calibration trials using 5, 10, 15, and 20 US-Camera acquisitions. For each calibration trial we randomly select a subset of the 20 acquisitions. Note that the calibration trials with 20 acquisitions always use the same input data. Since multiset-RANSAC is not deterministic, its results can be slightly different in each trial and thus we want to evaluate its repeatability.

In Fig. 4.7 we display, for all calibration trials, the estimation distributions in focal length, aspect ratio, translation norm, and rotation angle with respect to the $z-$axis of the calibration plane reference frame. We confirm our initial observations from the synthetic experiment that using less than 15 images leads to unstable results. It is also noteworthy that the overall results are worse than in simulation. The main reason is the presence of outlier line detections, which makes the problem

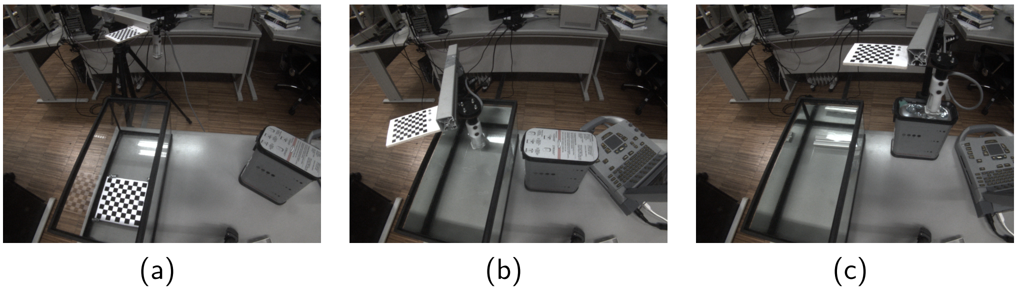
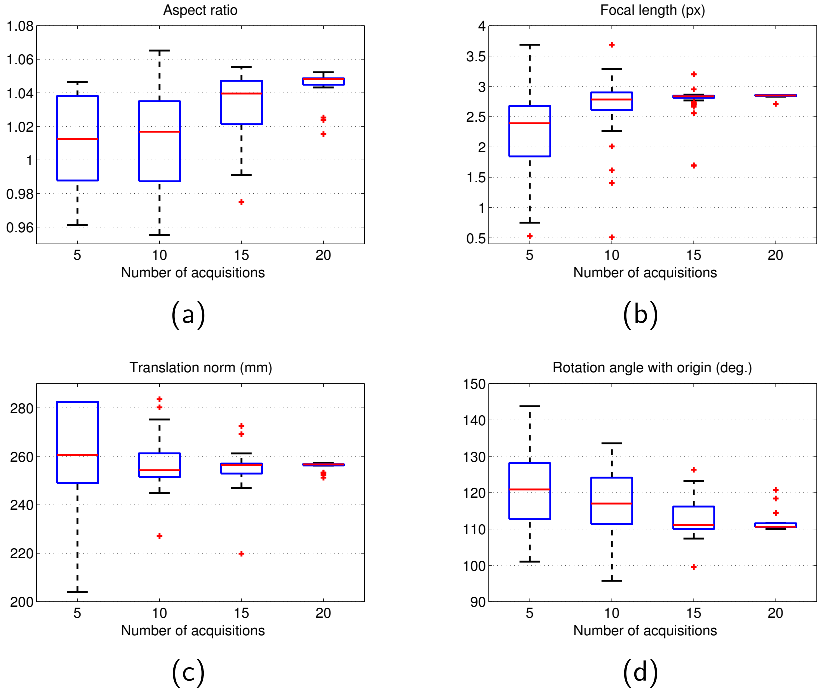


Figure 4.6: Experimental set-up: (a) Detection of the calibration plane (bottom of the empty tank) in the camera reference frame; (b) after filling the tank with water we perform US-Camera synchronized acquisitions; (c) Validation of calbration results is done with a precision phantom.

slightly more challenging. We are able to obtain stable calibration results with 20 images. This is a significant advance over the alternative plane-based methods that require hundreds of acquisitions in order to converge [70].

We evaluate the accuracy of our calibration with 20 acquisitions using a precision phantom that contains a set of horizontal parallel wires immersed in a tissue mimicking gel, such that their relative positions are accurately known. Note that in the phantom's gel, the sound travels 3% faster than in water, thus the focal length must be adjusted by the same factor in this validation.

In order to represent the wires in the reference frame of the calibration plane we position the phantom such that the wires are oriented along the $y-$axis of the calibration plane (as displayed in Figs. 4.6(a), 4.8(a)). Additionally we manually mark one of the wires in an US image (blue point in Fig. 4.8(a)) to define the origin of the phantom reference frame. Now we are able to represent the wires as 3D lines in the reference frame of the calibration plane (Fig. 4.8(b)). Finally the remaining wires are re-projected on the US image using our estimation of the extrinsic and intrinsic calibration (green points in Fig. 4.8(a)). The coincidence between the re-projected points and the US measurements shows that our calibration is accurate.



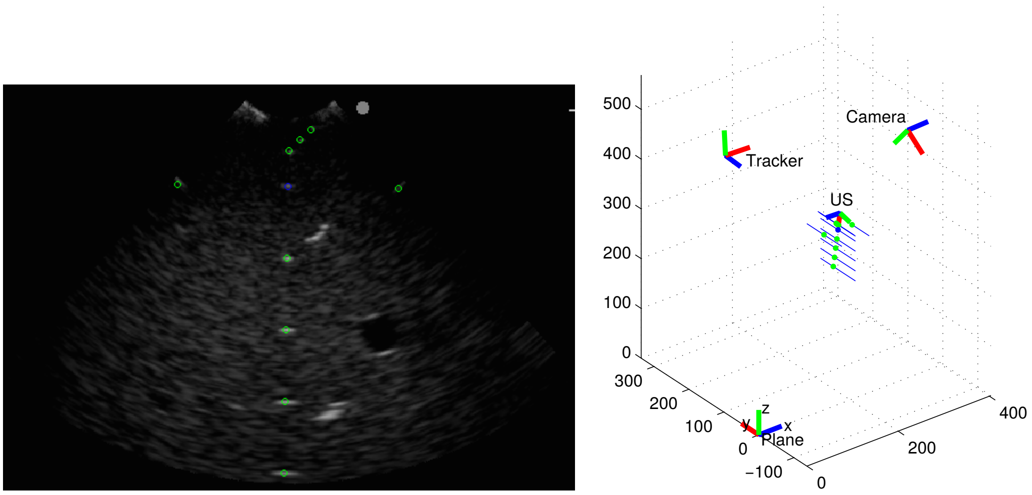
(a)

(b)

(c)

(d)

Figure 4.7: Distribution of the calibration parameters with real data. For each case we perform 50 trials.



(a) (b)

Figure 4.8: Phantom validation: (a) the blue point is manually marked to define the origin. The green points are re-projected according to the phantom specifications and a calibration with 20 acquisitions; (b) 3D representation of the validation setup. The blue lines are the phantom wires and the points are their intersections with the US cutting plane.

Chapter 5

Full camera calibration from independent pairwise correspondences

5.1 Introduction

Camera networks are sets of cameras whose fields of view are usually shared between two or more cameras within a network. They find applications in several domains that are concerned with the capture, the recording, and the analysis of dynamic scenes, for instance surveillance and animation modelling applications [75]. Most of these applications require the calibration of cameras in order to perform geometric operations such as reconstruction. One widely proposed appproach to achieve this purpose is to use a calibration pattern or rig [76–81], which is typically an offline procedure, requiring human intervention. However, there are many situations where a simpler and unsupervised scheme is desirable, in particular when adding a camera or modifying its location or characteristics while operating the network.

Without prior 3D information, image correspondences between cameras must be considered. Since camera networks are often sparse, correspondences in 3 or more images can be difficult to obtain, hence preventing the use of traditional calibration tools [37]. In contrast, correspondences between 2 images are more likely to be available by construction of camera networks.

This chapter addresses the issue of fully calibrating a camera given independent correspondences with 2 calibrated cameras. The situations particularly targeted are the addition or the modification of a camera in a calibrated network under operation,

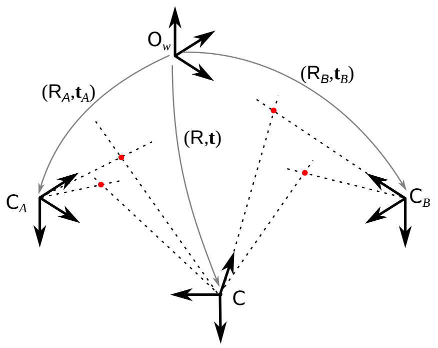

O_w
$(\mathsf{R}_A,\mathbf{t}_A)$
$(\mathsf{R}_B,\mathbf{t}_B)$
$(\mathsf{R},\mathbf{t})$
C_A
C_B
C

Figure 5.1: We consider the problem of fully calibrating the camera C, given pairwise correspondences with two calibrated cameras C_A and C_B.

which are common situations in practical camera network setups. The literature provides solutions when correspondences with 3 or more cameras are available [37], as well as when there is a mixture of corresponences with 2 and 3 cameras [82]. However, few efforts have been made to solve for the case with only 2-view correspondences. We investigate this issue and derive a minimal solution that requires 11 correspondences to estimate the 11 parameters of the unknown camera.

5.2 Problem Statement

Let us consider two calibrated cameras C_A and C_B, such that the matrices of intrinsic parameters are K_A and K_B, and the absolute poses are expressed in a world coordinate system $\mathbf{O}_w$ by the rotations matrices R_A and R_B, and the translation vectors $\mathbf{t}_A$ and $\mathbf{t}_B$. Consider an additional camera C for which both the intrinsic calibration K, and the extrinsic calibration R, $\mathbf{t}$ are unknown. Our article addresses the problem of calibrating this third camera using as input data a set of a image correspondences $(\mathbf{x}^{(i)}, \mathbf{x}_A^{(i)})$ between C and C_A, and set of b image correspondences $(\mathbf{x}^{(a+j)}, \mathbf{x}_B^{(j)})$ between C and C_B (Fig. 5.1). We assume that the two sets of pairwise matches are independent, meaning that

$$\mathbf{x}^i \neq \mathbf{x}^{a+j}, \, \forall_{i=1\ldots a,\, j=1\ldots b} \, .$$

In other words, there are no triplets of correspondences generated by scene points that are simultaneously seen by the three cameras. The absence of triple matches precludes the application of the standard calibration techniques that are described in text books [3, 37]. These approaches typically rely on the recovery of 3D points using the calibrated stereo views and standard triangulation [83]. These 3D points can in turn be used as reference points for the calibration of the 3rd camera (resection) [37, 77]. A possible alternative is to build a *measurement matrix* with the image correspondences, and perform projective factorization using the Sturm-Triggs algorithm [84] with a suitable extension for handling missing data [76]. However, this class of methods is meant for problems with multiple cameras and large number of correspondences, and it is unlikely that they will converge to a solution with pairwise correspondences only. One could also represent the camera network as a collection of fundamental matrices and obtain the trifocal tensor constraints with the new camera [85] and although this avoids an explicit 3D reconstruction of points, it still requires triple correspondences in order to calibrate the new node.

In summary, and to the best of our knowledge, the calibration of a camera using independent pairwise correspondences with two other views has never been addressed in the literature before. We present in the following a minimal solution when 7 or more matches with one of the views are available.

5.3 Linear system of equations with a minimum number of unknowns

In this section we derive a system of linear equations that has a minimum number of unknowns and fully constrains the camera calibration. The problem is formulated in the context of epipolar geometry between general camera models [18], with one side being the uncalibrated pin-hole camera C, and the other side being the pair of calibrated cameras C_A and C_B that can be understood as a particular instance of a non-central imaging device denoted by $\mathsf{C}_A \cup \mathsf{C}_B$. It is shown below that under such configuration the corresponding back-projection lines must satisfy a bilinear relation expressed by a 3×5 matrix, and that the estimation of the epipolar geometry using a DLT-like approach cannot be achieved with less than 14 pairwise matches.

Note that when the intrinsics are known, this problem is a particular case of the pose estimation between calibrated general camera models [18] that has already been solved both linearly [6] and using the minimal number of 6 pairwise correspondences [16].

5.3.1 Line Incidence Relations

Let $\mathbf{x}_A$ and $\mathbf{x}_B$ be image points in C_A and C_B. Since the cameras are fully calibrated, the corresponding back-projection lines $\mathbf{L}_A$ and $\mathbf{L}_B$ can be expressed in the common world reference frame $\mathbf{O}_w$ by a homogeneous Plücker vector

$$\mathbf{L}_{A/B} \sim \begin{pmatrix} \mathbf{d}_{A/B} \\ \mathbf{m}_{A/B} \end{pmatrix} ,$$

with the 3-vectors $\mathbf{d}_{A/B}$ and $\mathbf{m}_{A/B}$ being respectively the direction and the momentum of the line belonging to either camera A or B. In a similar manner, an image point $\mathbf{x}$ in C gives rise to a back-projection line $\mathbf{L}$ that is represented in the local camera reference frame by

$$\mathbf{L} \sim \begin{pmatrix} \mathbf{d} \\ \mathbf{0} \end{pmatrix} ,$$

with the direction depending on the matrix of intrinsic parameters K

$$\mathbf{d} \sim \mathsf{K}^{-1} \mathbf{x} . \tag{5.1}$$

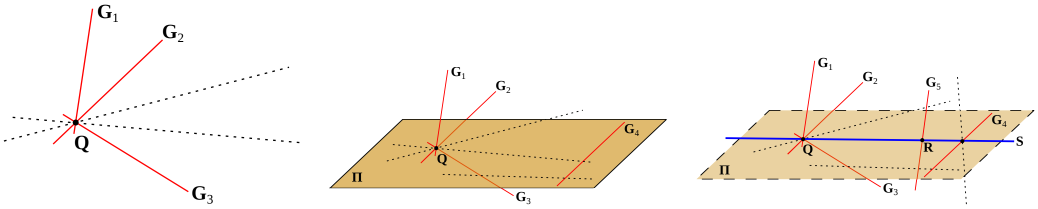
If $\mathbf{x}$ and $\mathbf{x}_{A/B}$ are image correspondences, then the back-projection lines $\mathbf{L}$ and $\mathbf{L}_{A/B}$ must be incident. Given the rigid displacement between the reference frames $\mathbf{O}_w$ and C, and the condition for two lines in Plücker coordinates to intersect, it comes that the following condition must hold

$$\mathbf{L}^\mathsf{T} \begin{pmatrix} 0 & \mathtt{I} \\ \mathtt{I} & 0 \end{pmatrix} \begin{pmatrix} \mathsf{R} & 0 \\ [\mathbf{t}]_\times \mathsf{R} & \mathsf{R} \end{pmatrix} \mathbf{L}_{A/B} = 0 .$$

Since the momentum of $\mathbf{L}$ is always zero, then the above equation can be re-written as

$$\mathbf{d}^\mathsf{T} \begin{pmatrix} [\mathbf{t}]_\times \mathsf{R} & \mathsf{R} \end{pmatrix} \mathbf{L}_{A/B} = 0 \tag{5.2}$$

Equation 5.2 is the particular case of the generalized epipolar constraint proposed in [18] when one of the cameras is a conventional pin-hole. However, and similarly to the general case, the bilinear relation between back-projection lines is expressed by a 3 × 6 matrix that encodes the calibration parameters. Therefore, the linear estimation of the 18 entries of the matrix up to a global scale factor still requires a minimum of 17 image correspondences between C and the camera pair $\mathsf{C}_A \cup \mathsf{C}_B$.



(a) Line Bundle

(b) Linear line subspace for $N = 4$

(c) Linear line congruent

Figure 5.2: Line subspaces; in each figure, the dotted lines represent possible lines that could result from the linear combination of the generation basis.

5.3.2 Analysis using linear line subspaces

In our case the parametrization of equation 5.2 leads to a linear estimation problem that is sub-determined. This is a situation similar to the degenerate configurations recently reported in [6] in the context of motion estimation using a calibrated multiset-camera rig. We use the theory of linear line subspaces [86] to explain the underlying reasons of the sub-determination, and prove that the calibration problem can be formulated in a linear manner using a minimum of 15 parameters.

It is well known that a line in 3D represented in Plücker coordinates can be thought as a point in $\mathbb{P}^5$ lying in the so-called *Klein Quadric*. Let us consider a generic hyperplane in $\mathbb{P}^5$ with dimension $N \leq 5$. The hyperplane intersects the quadric in a locus that defines, via Plücker mapping, a certain subset of lines in the original 3D space. This subset is called a *linear line subspace* (LLS) of dimension N, and each line $\mathbf{L}$ in the LLS is in the linear span of N other lines $\mathbf{G}$ with independent Plücker vectors [86].

The lines going through a generic 3D point $\mathbf{Q}$ form a *line bundle* that is often used to model the back-projection rays of a pin-hole camera (Fig. 5.2(a)). The line bundle is a LLS of dimension $N = 3$, and each line $\mathbf{L}$ going through $\mathbf{Q}$ can be uniquely expressed as the linear combination of any other three non-coplanar lines in the bundle. It is said that these three lines $\{\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3\}$ are a basis for the LLS. Consider now an additional line $\mathbf{G}_4$ that is not in the bundle. In this case the span of $\{\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, \mathbf{G}_4\}$ contains a LLS of dimension $N = 4$ that comprises all the lines that go through point $\mathbf{Q}$, and all the lines that lie in the plane $\mathbf{\Pi}$ defined by $\mathbf{Q}$ and $\mathbf{G}_4$ (Fig. 5.2(b)). Finally, the addition of a fifth line $\mathbf{G}_5$ to the basis gives raise to a LLS of dimension $N = 5$ that is called a *linear line congruent* (LLC) [86]. The LLC includes all the lines intersecting an axis $\mathbf{S}$ that, in the particular case of Fig. 5.2(c), is defined by the center $\mathbf{Q}$ and the point $\mathbf{R}$ where $\mathbf{G}_5$ meets the plane $\mathbf{\Pi}$.

Let us return to our calibration problem where the generalized camera $\mathsf{C}_A \cup \mathsf{C}_B$ is modeled by the union of two distinct line bundles. The key observation is that every possible back-projection line $\mathbf{L}_{A/B}$ must be tangent to the line going through C_A and C_B (the baseline). Thus, and since the lines $\mathbf{L}_{A/B}$ are contained in a LLC, they can be represented in a unique manner as the linear combination of any 5 lines $\mathbf{G}_i$ that intersect the baseline

$$\mathbf{L}_{A/B} \sim \underbrace{\begin{pmatrix} \mathbf{G}_1 & \mathbf{G}_2 & \mathbf{G}_3 & \mathbf{G}_4 & \mathbf{G}_5 \end{pmatrix}}_{\mathsf{G}} \boldsymbol{\lambda}_{A/B} \, ,$$

where G is a 6×5 matrix with full rank, and $\boldsymbol{\lambda}_{A/B}$ is a 5-vector defined up to scale. Replacing in equation 5.2 yields

$$\mathbf{d}^{\mathsf{T}} \begin{pmatrix} [\mathbf{t}]_{\times}\mathsf{R} & \mathsf{R} \end{pmatrix} \mathsf{G} \, \boldsymbol{\lambda}_{A/B} = 0 \tag{5.3}$$

We have just re-written the epipolar constraint of equation 5.2 as a bilinear relation between the direction $\mathbf{d}$ of the line $\mathbf{L}$ in camera C, and the representation $\boldsymbol{\lambda}_{A/B}$ of the back-projection line $\mathbf{L}_{A/B}$ in the generalized camera $\mathsf{C}_A \cup \mathsf{C}_B$. Since the bilinear relation is now encoded by a 3×5 matrix with 15 entries, then 14 image point correspondences are sufficient for estimating the epipolar geometry in a DLT-like manner. The discussion clearly explains why the 18 parameter formulation of equation 5.2 is ambiguous [6], and shows that a compact linear formulation of the stated calibration problem must necessarily have 15 parameters because the lowest dimensional linear sub-space containing all the back-projection rays of two pin-holes is a LLC.

5.3.3 Compact linear formulation

Given the two arbitrary calibrated cameras, it is always possible to perform a change of reference frames for achieving the configuration exhibited in Fig. 5.3. We consider, without any loss of generality, that the world reference frame is aligned with the coordinate system of camera C_A, and that the X-axis is coincident with the baseline defined by the projection centers of the two pin-holes. The local reference frame of the second camera is assumed to have origin in C_B and to be parallel to the coordinate system of C_A. Under such circumstances the rigid transformation that

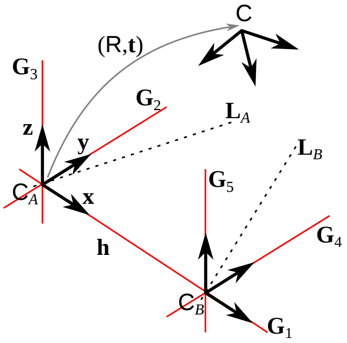


Figure 5.3: The space generated by two bundles of lines (the rays of 2 pinhole cameras) can be fully represented as the linear span of $\{\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, \mathbf{G}_4, \mathbf{G}_5\}$.

maps point coordinates from C_B to C_A is given by

$$\mathsf{T}_{B\rightarrow A} = \begin{pmatrix} \mathtt{I} & \mathbf{h} \\ \mathbf{0} & 1 \end{pmatrix}$$

with $\mathtt{I}$ being the 3×3 identity matrix and $\mathbf{h} = \begin{pmatrix} h & 0 & 0 \end{pmatrix}^\mathsf{T}$. Since the axes X, Y, Z of the system of coordinates of C_A, and the axes Y, Z of the reference frame of C_B are linearly independent lines, then they can be used to establish a basis G for the LLC defined by the baseline. It comes that

$$\mathsf{G} \sim \begin{pmatrix} \mathtt{I} & \mathtt{I}^{\{2,3\}} \\ \mathsf{0} & [\mathbf{h}]_\times^{\{2,3\}} \end{pmatrix}$$

with the upper script $\{2, 3\}$ denoting the second and third columns of the matrix.

Let us now consider an image correspondence $(\mathbf{x}, \mathbf{x}_A)$ between C and C_A. The back-projection of $\mathbf{x}_A$ is a line $\mathbf{L}_A$ with direction $\mathbf{d}_A$ expressed in the reference frame of C_A. Given the basis G above, it comes that $\mathbf{L}_A \sim \mathsf{G}\,\boldsymbol{\lambda}_A$ with

$$\boldsymbol{\lambda}_A \sim \begin{pmatrix} \mathbf{d}_A^\mathsf{T} & 0 & 0 \end{pmatrix}^\mathsf{T}.$$

Replacing in equation 5.3, and making $\mathbf{d} \sim \mathsf{K}^{-1}\,\mathbf{x}$, yields

$$\mathbf{x}^\mathsf{T}\,\mathsf{F}_A\,\mathbf{d}_A = 0 \tag{5.4}$$

with F_A being the standard fundamental matrix between the uncalibrated camera C and the calibrated view C_A

$$\mathsf{F}_A \;=\; \mathsf{K}^{-\mathsf{T}}\,[\mathbf{t}]_\times\,\mathsf{R} \tag{5.5}$$

Repeating the reasoning for the case of an image correspondence $(\mathbf{x}, \mathbf{x}_B)$ between C and C_B, it comes that

$$\boldsymbol{\lambda}_B \;\sim\; \begin{pmatrix} d_{B,1} & 0 & 0 & d_{B,2} & d_{B,3} \end{pmatrix}^\mathsf{T}.$$

with $\mathbf{d}_B \;\sim\; \begin{pmatrix} d_{B,1} & d_{B,2} & d_{B,3} \end{pmatrix}^\mathsf{T}$ being the direction of the back-projection line $\mathbf{L}_B$ expressed in the local reference frame of camera C_B. Making $\mathbf{L}_B \;\sim\; \mathsf{G}\,\boldsymbol{\lambda}_B$ in equation 5.3, and taking into account that the first column of $[\mathbf{h}]_\times$ is a null vector, we obtain that

$$\mathbf{x}^\mathsf{T}\,\mathsf{F}_B\,\mathbf{d}_B \;=\; 0 \tag{5.6}$$

with F_B being the fundamental matrix between C and C_B that can be written as

$$\mathsf{F}_B \;=\; \mathsf{F}_A + \mathsf{K}^{-1}\mathsf{R}[\mathbf{h}]_\times. \tag{5.7}$$

It follows from the equation above that the first columns of matrices F_A and F_B are always equal ($\mathsf{F}_A^{\{1\}} \;=\; \mathsf{F}_B^{\{1\}}$).

Given the image correspondences $(\mathbf{x}^{(i)}, \mathbf{x}_A^{(i)})$, with $i = 1, \ldots a$, and $(\mathbf{x}^{(a+j)}, \mathbf{x}_B^{(j)})$ with $j = 1, \ldots b$, we can determine the line directions $\mathbf{d}_A^{(i)} \;\sim\; \mathsf{K}_A^{-1}\,\mathbf{x}_A^{(i)}$ and $\mathbf{d}_B^{(j)} \;\sim\; \mathsf{K}_B^{-1}\,\mathbf{x}_B^{(j)}$, and establish a system of linear equations (equation 5.8) based on the bilinear constraints of equations 5.4 and 5.6.

$$\begin{pmatrix}
x_1^{(1)}\mathbf{d}_A^{(1)\mathsf{T}} & x_2^{(1)}\mathbf{d}_A^{(1)\mathsf{T}} & x_3^{(1)}\mathbf{d}_A^{(1)\mathsf{T}} & \mathsf{0}^\mathsf{T} & \mathsf{0}^\mathsf{T} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
x_1^{(a)}\mathbf{d}_A^{(a)\mathsf{T}} & x_2^{(a)}\mathbf{d}_A^{(a)\mathsf{T}} & x_3^{(a)}\mathbf{d}_A^{(a)\mathsf{T}} & \mathsf{0}^\mathsf{T} & \mathsf{0}^\mathsf{T} \\
x_1^{(a+1)}\mathbf{d}_B^{(1)\mathsf{T}} & \mathsf{0}^\mathsf{T} & \mathsf{0}^\mathsf{T} & x_2^{(a+1)}\mathbf{d}_B^{(1)\mathsf{T}} & x_3^{(a+1)}\mathbf{d}_B^{(1)\mathsf{T}} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
x_1^{(a+b)}\mathbf{d}_B^{(b)\mathsf{T}} & \mathsf{0}^\mathsf{T} & \mathsf{0}^\mathsf{T} & x_2^{(a+b)}\mathbf{d}_B^{(b)\mathsf{T}} & x_3^{(a+b)}\mathbf{d}_B^{(b)\mathsf{T}}
\end{pmatrix}
\begin{pmatrix} \mathsf{F}_A^{\{1\}} \\ \mathsf{F}_A^{\{2\}} \\ \mathsf{F}_A^{\{3\}} \\ \mathsf{F}_B^{\{2\}} \\ \mathsf{F}_B^{\{3\}} \end{pmatrix} = 0 \tag{5.8}$$

If $a + b \geq 14$ then the fundamental matrices F_A and F_B can be determined up to a common scale factor using a standard DLT approach.

5.4 A minimal solution for the estimation of F_A and F_B

We have shown that the two fundamental matrices, F_A and F_B, that encode the calibration information K, R, and $\mathbf{t}$, can be determined from a minimum of 14 independent image correspondences. However, the total number of independent unknowns is 11 (5 intrinsic parameters and 6 extrinsic parameters) meaning that the estimation problem can be further constrained. Two of these constrains are rather obvious:

$$\det(\mathsf{F}_A) = 0 \tag{5.9}$$

$$\det(\mathsf{F}_B) = 0 \tag{5.10}$$

For the third constraint it must be observed that the sum of F_A and F_B is still a fundamental matrix. From equations 5.5 and 5.7 it comes after algebraic manipulation that

$$\mathsf{F}_A + \mathsf{F}_B = \mathsf{K}^{-1}[2\mathbf{t} + \mathsf{R}\mathbf{h}]_\times \mathsf{R}\,,$$

which means that the following condition must hold

$$\det(\mathsf{F}_A + \mathsf{F}_B) = 0 \tag{5.11}$$

The equation above basically enforces the condition that F_A and F_B must be two fundamental matrices encoding the same rotation R.

5.4.1 Outline of the estimation algorithm

This section outlines an algorithm for estimating F_A and F_B from pairwise correspondences between an uncalibrated camera C and two calibrated cameras C_A , C_B. Each pairwise correspondence contains a homogeneous point $\mathbf{x}$ in pixel coordinates belonging to C and a homogeneous point $\mathbf{d}_{A/B}$ in metric coordinates belonging to either C_A or C_B. Consider as input the correspondences $(\mathbf{x}^{(i)}, \mathbf{d}_A^{(i)})$, with $i = 1, \dots a$, and $(\mathbf{x}^{(a+j)}, \mathbf{d}_B^{(j)})$ with $j = 1, \dots b$, where $a + b \geq 11$. The solution is found by determining the 4-dimensional null space of the measurement matrix of the the linear system of equation 5.8, followed by intersecting the span of this null-space with the locus defined by the polynomials constraints of equations 5.9 to 5.11. Instead of solving a system of 3 third order polynomials in 3 variables, we explore the sparsity of the measurement matrix and simplify the problem to solving 1 cubic polynomial in 1 variable, and a system of 2 quadratic polynomials in 2 variables. In order for this to be possible 7 of the 11 image matches must be in the same calibrated view ($a = 7$, $b = 4$).

1. Build the linear system of equation 5.8 from the 11 pairwise correspondences, and determine the 4-dimensional solution space using SVD decomposition ($\mathsf{A} = \mathsf{USV}^\mathsf{T}$). The solution space is spanned by the columns of the 15×4 matrix $\mathsf{V}^{\{12 \dots 15\}}$ (the last 4 columns of V)

2. The first 9 rows of $\mathsf{V}^{\{12 \dots 15\}}$ define always a rank 2 sub-matrix due to the structure of the linear system and the fact that $a = 7$. Thus, the solution space of F_A is spanned by the two columns of the sub-matrix, $\mathbf{a}$ and $\mathbf{a}'$, that are linearly independent, which enables to write $\mathsf{F}_A(\alpha) = \mathsf{A}' + \alpha\mathsf{A}$ with α being a free parameter.

3. Compute α by solving the cubic constraint of equation 5.9 and determine the fundamental matrix F_A.

4. Substitute F_A in the linear system which results in 4 equations in 7 unknowns. The solution space of this system is 3-dimensional and F_B can be written as the linear span $\mathsf{F}_B(\beta_1, \beta_2) = \mathsf{B}'' + \beta_1\mathsf{B}' + \beta_2\mathsf{B})$

5. Substitute F_A and $\mathsf{F}_B(\beta_1, \beta_2)$ in equations 5.10 and 5.11. This leads to a bivariate system of 2 quadratic equations that corresponds geometrically to determining the point intersections of two conic curves. Compute β_1 and β_2 by solving the bivariate system [87], and determine the fundamental matrix F_B.

Since the cubic equation of step 3 gives up to 3 discrete solutions, and the bivariate system of quadric equations has at most 4 distinct solutions, then there is a maximum of 12 possible solutions for the pair of fundamental matrices $(\mathsf{F}_A, \mathsf{F}_B)$.

5.5 Factorization of F_A and F_B

So far we have shown how to estimate the fundamental matrices F_A and F_B from a minimum of 11 pairwise correspondences. In order to solve the calibration problem, F_A and F_B must be factorized into the intrinsic parameters K and the relative pose R, $\mathbf{t}$. The absence of intrinsics in the right side of the fundamental matrices leads to a simplified version of Kruppa's equations [3, 37] that enable the recovery of K in a relatively straightforward manner. This section discusses how this can be

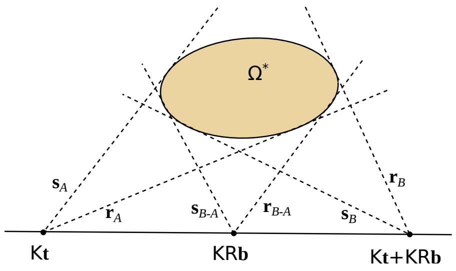


Figure 5.4: Conic envelope Ω establishes linear relations $\mathbf{s}^{\mathsf{T}}\mathsf{K}\mathsf{K}^{\mathsf{T}}\mathbf{s} = 0$ and $\mathbf{r}^{\mathsf{T}}\mathsf{K}\mathsf{K}^{\mathsf{T}}\mathbf{r} = 0$.

achieved. After knowing K, we can compute the essential matrix E_A and apply standard techniques for determining the rotation R and the translation $\mathbf{t}$ up to scale factor [3,37]. This scale factor can be easily found using the known baseline between C_A and C_B.

Let us now discuss the extraction of the matrix K. Consider the fundamental matrix F_A that is given in equation 5.5. After some algebraic manipulations we obtain that

$$\mathsf{F}_A\,\mathsf{F}_A^{\mathsf{T}} \sim [\mathsf{K}\mathbf{t}]_\times\,\mathsf{K}\mathsf{K}^{\mathsf{T}}\,[\mathsf{K}\mathbf{t}]_\times$$

From the result above it follows that, if $\mathbf{y}$ is a point in the projective plane that satisfies

$$\mathbf{y}^{\mathsf{T}}\,\mathsf{F}_A\,\mathsf{F}_A^{\mathsf{T}}\,\mathbf{y} = 0\,,$$

then the line defined by $\mathbf{y}$ and the left epipole of F_A must lie in the conic envelope $\mathsf{K}\mathsf{K}^{\mathsf{T}}$ that is the dual of the image of the absolute conic (DIAC) [3,37]. $\mathsf{F}_A\,\mathsf{F}_A^{\mathsf{T}}$ is a rank 2 symmetric matrix that represents a degenerate conic. Thus, and since this conic consists in two lines $\mathbf{s}_A$, $\mathbf{r}_A$ that intersect in the left epipole, it is easy to conclude that $\mathbf{s}_A$, $\mathbf{r}_A$ must belong to the DIAC. The same reasoning can be applied to the fundamental matrix F_B of equation 5.7

$$\mathsf{F}_B\,\mathsf{F}_B{}^{\mathsf{T}} \sim [\mathsf{K}(\mathsf{R}\mathbf{h}+\mathbf{t})]_\times\mathsf{K}\mathsf{K}^{\mathsf{T}}[\mathsf{K}(\mathsf{R}\mathbf{h}+\mathbf{t})]_\times\,,$$

and to the matrix $\mathsf{F}_B - \mathsf{F}_A$ that is still rank deficient because the first columns of the two fundamental matrices are equal

$$(\mathsf{F}_B - \mathsf{F}_A)(\mathsf{F}_B - \mathsf{F}_A)^{\mathsf{T}} \sim [\mathsf{K}\mathsf{R}\mathbf{h}]_\times\mathsf{K}\mathsf{K}^{\mathsf{T}}[\mathsf{K}\mathsf{R}\mathbf{h}]_\times\,.$$

Summarizing, and as shown in Fig. 5.4, the DIAC is fully constrained by the line pairs arising from the rank 2 degenerate conics $\mathsf{F}_A\mathsf{F}_A^\mathsf{T}$, $\mathsf{F}_B\mathsf{F}_B^\mathsf{T}$, and $(\mathsf{F}_B - \mathsf{F}_A)(\mathsf{F}_B - \mathsf{F}_A)^\mathsf{T}$. It is important to note that, although we have six lines, they only give rise to five independent constraints on the parameters of the DIAC. This is explained by the fact that their pairwise intersections are collinear.

Another possible factorization approach would be to use the properties of the SVD decomposition of a fundamental matrix [88].

5.6 Practical considerations

5.6.1 Multiset-MAPSAC

We now consider this calibration problem in a practical scenario, where in general there are more than 11 correspondences, some of them being outliers. We have two datasets: one contains K_A correspondences between C and C_A, and the other K_B correspondences between C and C_B.

This fits into our multiset-dataset framework from Chapter 2, however, there are some particularities that must be discussed. Unlike the multiset-RANSAC application in Chapter 4, in this case there are 2 available datasets and the minimal solution requires both of them to generate a candidate solution. Pairwise correspondences with one camera would just give us a fundamental matrix, while the pairwise correspondences with two cameras give us both the extrinsic and intrinsic camera calibration. This means that a candidate model with many inliers in one dataset and very few on the other is a poor solution that is over-fitting to a particular fundamental matrix. To tackle this issue we use the multiset-MAPSAC approach and define a prior probability function that penalizes significantly uneven distributions of inliers.

To use multiset-MAPSAC we generate candidate solutions by first randomly selecting from which of the two datasets we sample 7 and 4 correspondences, and then randomly selecting the respective number of samples from each dataset.

The error of a candidate solution is measured in terms of the perpendicular distances between point correspondences and their epipolar lines. Given a pairwise point correspondence $(\mathbf{x}, \widehat{\mathbf{x}})$ between two cameras related by a fundamental matrix F, the epipolar error r is the distance in pixels between point $\mathbf{x}$ and the epipolar line of $\widehat{\mathbf{x}}$

$$r = \frac{\mathbf{x}^\mathsf{T}\mathsf{F}^\mathsf{T}\widehat{\mathbf{x}}}{||\mathtt{I}_{2\times 3}\mathsf{F}^\mathsf{T}\widehat{\mathbf{x}}||}. \tag{5.12}$$

Analogously, the distance between point $\widehat{\mathbf{x}}$ and the epipolar line of $\mathbf{x}$ is

$$\widehat{r} = \frac{\widehat{\mathbf{x}}^{\mathsf{T}}\mathsf{F}\mathbf{x}}{||\mathtt{I}_{2\times 3}\mathsf{F}\mathbf{x}||}. \tag{5.13}$$

Note that in this context the F is the standard fundamental matrix between two views and not F_A or F_B.

In each iteration we estimate the inlier ratios of each dataset γ_A, γ_B with expectation maximization, and then the multiset-MAPSAC cost function is given by the posterior probability

$$c = Pr(\gamma_A, \gamma_B)L(\mathsf{F}_A, \mathsf{F}_B|r_1, \widehat{r}_1, ..., r_{(K_A+K_B)}\widehat{r}_{(K_A+K_B)}) \tag{5.14}$$

The likelihood term is defined by equation 2.9 from Chapter 2 (multiset-MLESAC). As a rule of thumb, we assume a Gaussian variance of 1 pixel for inliers, and an outlier range equal to the diagonal length of each image in pixels. The prior term $Pr(\gamma_A, \gamma_B)$ is a probability density function that penalizes uneven distributions of inliers

$$Pr(\gamma_A, \gamma_B) = (\alpha + 1)^2(\gamma_A\gamma_B)^\alpha \tag{5.15}$$

where α is set to a value with the same order of magnitude as the number of correspondences in each dataset. Note that the constant factor $(\alpha + 1)^2$ is just to guarantee that $Pr(\gamma_A, \gamma_B)$ is a probability density function for γ_A, γ_B between 0 and 1. In our context of maximum a posteriori estimation it can be ignored.

As stated in Chapter 2, when the minimal solution requires all available datasets, some additional concerns should be taken into account in order to compute the number of multiset-MAPSAC iterations. In this case, equation 2.4 is not suitable to compute the probability p_{inall} of sampling only inliers in each iteration. In the dataset sampling step there are only two possible outcomes: choosing from which dataset we select 4 and 7 correspondences. Therefore, the probability of selecting only inliers in this process is

$$p_{inall} = \frac{1}{2}(\gamma_A^7\gamma_B^4 + \gamma_A^4\gamma_B^7) \tag{5.16}$$

The multiset-MAPSAC formulation can also be extended to a scenario where we have correspondences with N cameras C_1,...,C_N. In this case the multiset-MAPSAC first samples 2 random datasets and then extracts 4 and 7 correspondences from each of them. Note, however, that the over-fitting case discussed above can only happen when a candidate solution fits well into just one dataset. Thus, the prior term $Pr(\gamma_A, \gamma_B)$ should be computed using only the two highest values from γ_1,...,γ_N.

5.6.2 Pre-filtering

Using multiset-MAPSAC with our calibration method raises another important practical issue. The minimal solution for our problem requires 11 points, a very high number that can lead to a huge computational cost in the presence of datasets highly contaminated by outliers.

With very challenging datasets the number of iterations can be drastically reduced with a pre-filtering stage. Each set of pairwise correspondences must verify a 2-view constraint with respectively F_A and F_B. This means that the 7-point algorithm can be used within RANSAC to eliminate outliers on both sets of correspondences. Note, however, that we should not use these independent estimations for F_A and F_B, since the constraint $\det(\mathsf{F}_A + \mathsf{F}_B) = 0$ is neglected. It is also likely that some outlier correspondences still remain after this step. In the experimental section of this chapter with real data, outlier ratios after pre-filtering range between 0% and 10%. Therefore, this pre-filtering stage should only be used to remove a first round of outliers, while in a second stage, muli-MAPSAC is used with the 11-point algorithm to find a consistent solution to F_A and F_B on datasets with much lower ratios of outliers.

5.6.3 Iterative least-squares refinement

A final refinement with bundle adjustment should be performed to achieve an optimal solution. Usually bundle adjustment minimizes the re-projection of reconstructed 3D points onto the cameras. However, since our formulation only uses pairwise correspondences, the introduction of unknown 3D points is an unnecessary burden. As described in [37], an explicit representation of 3D points can be avoided by minimizing the perpendicular distances between point correspondences and their epipolar lines.

Given a pairwise point correspondence $(\mathbf{x}, \widehat{\mathbf{x}})$ between two cameras related by a fundamental matrix F, the epipolar error r is defined by equations 5.12 and 5.13. We now consider a network with M calibrated cameras with rotations $\{\mathsf{R}_1, \mathsf{R}_2, ..., \mathsf{R}_M\}$, translations $\{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_M\}$, and intrinsics $\{\mathsf{K}_1, \mathsf{K}_2, ..., \mathsf{K}_M\}$ in a common reference frame, and a new camera with unknown parameters R, $\mathbf{t}$, K. The new camera has a set of N_j pairwise correspondences $\{(\mathbf{x}_{j,1}, \widehat{\mathbf{x}}_{j,i}), (\mathbf{x}_{j,2}, \widehat{\mathbf{x}}_{j,2}), ..., (\mathbf{x}_{j,N_j}, \widehat{\mathbf{x}}_{j,N_j})\}$ with each calibrated camera $j = 1, 2, ...M$. Therefore, the bundle adjustment problem becomes

$$\min_{\mathsf{R},\mathbf{t},\mathsf{K}} \sum_{j=1}^{M} \sum_{i=1}^{N_j} r_{j,i}^2 + \widehat{r}_{j,i}^2 \tag{5.17}$$

with

$$r_{j,i} = \left(\frac{\mathbf{x}_{j,i}{}^{\mathsf{T}} \mathsf{F}_j{}^{\mathsf{T}} \widehat{\mathbf{x}}_{j,i}}{||\mathtt{I}_{2\times3} \mathsf{F}_j{}^{\mathsf{T}} \widehat{\mathbf{x}}_{j,i}||} \right) \tag{5.18}$$

$$\widehat{r}_{j,i} = \left(\frac{\widehat{\mathbf{x}}_{j,i}^{\mathsf{T}} \mathsf{F}_j \mathbf{x}_{j,i}}{||\mathtt{I}_{2\times3} \mathsf{F}_j \mathbf{x}_{j,i}||} \right) \tag{5.19}$$

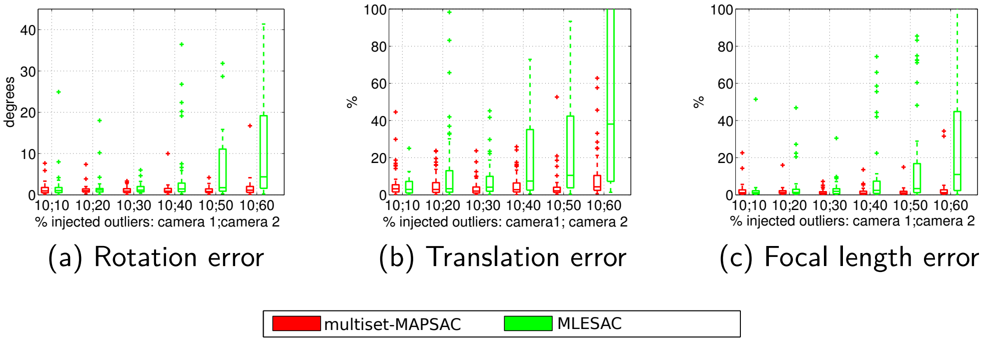
$$\mathsf{F}_j = \mathsf{K}_j [\mathsf{R}_j{}^{\mathsf{T}} \mathbf{t} + \mathbf{t}_j]_\times \mathsf{R}_j{}^{\mathsf{T}} \mathsf{R} \mathsf{K}. \tag{5.20}$$

5.7 Experiments

In this section we validate our calibration method using both synthetic data and real imagery of dynamic scenes acquired in a camera network environment. Real data was acquired with the Grimage platform [89], a room with a set of calibrated cameras that acquires synchronized frames of the same scene.

In a first set of experiments we use synthetic data to demonstrate that in challenging scenarios our multiset-MAPSAC formulation is essential to obtain accurate calibrations. We then demonstrate the usefulness of our calibration method in practice. We present two camera network applications where pairwise correspondences are significantly more abundant than triple correspondences, and therefore our algorithm outperforms competing approaches that require correspondences with more than two views. The first one is the addition of a new camera to a calibrated network, using a synchronized set of frames at a single time instant. In this case we compare our method against the 6-point approach that requires triple correspondences. In the second scenario the two methods are compared while performing the full calibration of a hand-held camera. Both its intrinsic parameters and its trajectory are estimated while capturing a dynamic scene within the field of view of the calibrated network.

We use SIFT features to establish point correspondences between the images. For both our method and the 6-point approach we perform a pre-filtering step with 7-point fundamental matrix estimation. For our method we use multiset-MAPSAC and the bundle adjustment described in section 5.6.3. On the other hand, the 6-



(a) Rotation error

(b) Translation error

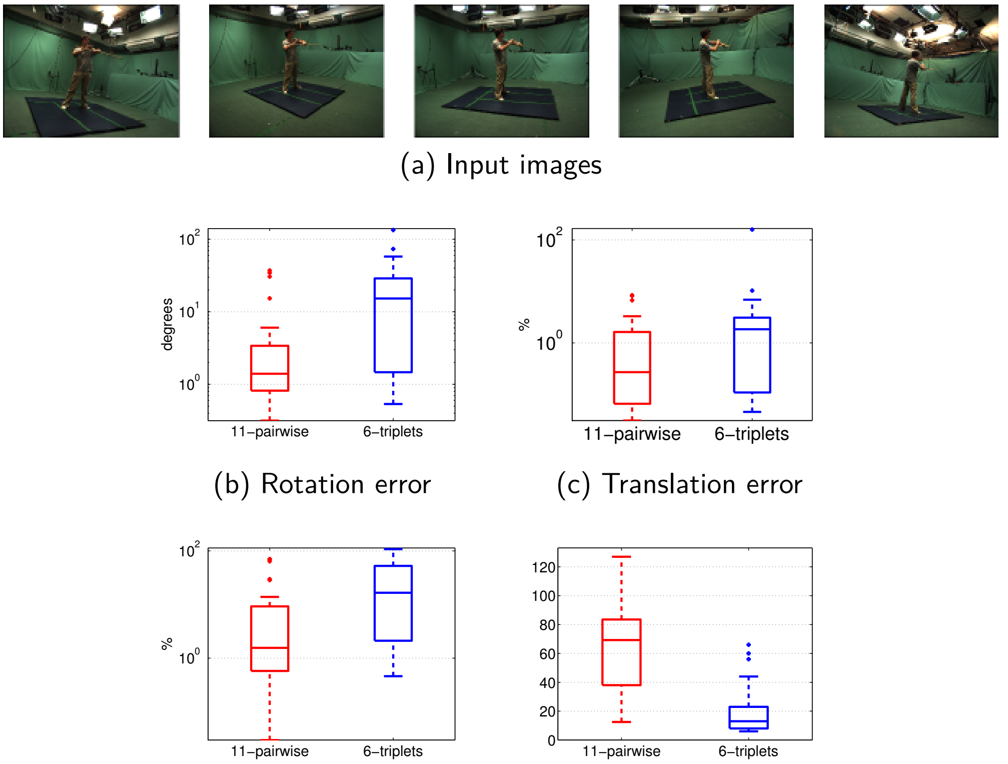
(c) Focal length error

Figure 5.5: Comparison between multiset-MAPSAC (red) and MLESAC (green) with synthetic data. Error distributions over 50 calibration trials for different levels of injected outliers.

point approach is a single dataset formulation and relies on 3D point estimation, therefore we use the standard versions of both MLESAC and bundle adjustment.

5.7.1 Validation of multiset-MAPSAC

We built a simulated environment in order to show that in some conditions, our multiset-MAPSAC formulation clearly outperforms a standard MLESAC approach that assumes all correspondences belong to the same dataset. Note that comparing multiset-MAPSAC against standard MAPSAC instead of MLESAC would not make any difference in the context of this problem since our defined prior probabilities depend exclusively on the assumption that different datasets have different inlier ratios. We generate calibrated cameras C_A, C_B, and an uncalibrated camera C in random poses such that they share a common field of view. Then we generate 500 points that are viewed by cameras C_A and C and 100 points that are viewed by cameras C_B and C. All these correspondences are injected with Gaussian noise with 1 pixel standard deviation, and also a predefined ratio of outliers that are random points following a uniform distribution within the image limits. We tried to calibrate camera C using the 11-point algorithm with both multiset-MAPSAC and MLESAC. We performed 50 calibration trials for each of six different levels of injected outliers in cameras C_A and C_B. In Fig 5.5 we show the error distributions for rotation, translation, and focal length when compared against groundtruth values. It is clear that multiset-MAPSAC is able to perform better in situations where the inlier ratios are significantly different in cameras C_A and C_B.



(a) Input images

(b) Rotation error

(c) Translation error

(d) Focal length error

(e) Number of inliers

Figure 5.6: Addition of a new node to a camera network. In each trial we try to calibrate one of the cameras in (a) assuming that the remaining four are calibrated. (b), (c), (d), (e) show the comparative performance between 11-pairwise and 6-triplets for 250 calibration trials

5.7.2 Addition of a new node to a calibrated network

In this experiment we aim at fully calibrating a camera using pairwise correspondences from a set of synchronized frames at a single time instant. For this purpose we used the *stick* dataset from the 4drepository [90], which contains synchronized frames from different calibrated cameras. We selected five cameras from a particular frame (Fig. 5.6(a)) and tried to calibrate each of them assuming we know the calibration of the remaining four cameras.

Our method presupposes that pairwise correspondences are only established with other two calibrated cameras. Therefore, we perform the pre-filtering step with the four calibrated cameras and select the two with the highest number of inliers. In the case of the 6-point algorithm all pre-filtered triple correspondences from the four cameras are used. Note, however, that there is a wide baseline between the five cameras, so typically the two closest cameras will produce the majority (if not all) the reliable correspondences.

After the pre-filtering step, and for each of the five cameras, both 6-triplets and 11-pairwise are run 50 times, summing up to 250 calibration tests for each approach. The error distributions for all calibration attempts are provided in Figs. 5.6(b),5.6(c),5.6(d), showing that our algorithm provides more accurate results. This can be explained by the fact that it is possible to establish a much higher number of pairwise correspondences than triple correspondences (Fig. 5.6(e)), despite the fact that triple correspondences are established across the four calibrated cameras, while for our algorithm we only use the pairwise correspondences from two cameras.

5.7.3 Calibration of a hand-held camera

We acquired a set of synchronized video sequences with both the calibrated network and a hand-held camera. It is composed of 30 frames in which the hand-held camera shares its field of view with two other calibrated cameras (Fig. 5.7). The viewed scene is dynamic, therefore there are only correspondences between images acquired at the same time instant. The intrinsic parameters of the hand-held camera were previously determined using a calibration target. We use these values as groundtruth for comparison with our estimates.

Both the intrinsic parameters and the trajectory of the hand-held camera can be recovered with our calibration method. In a first step, we calibrate each frame independently using pairwise correspondences with the synchronized frames from the calibrated cameras. A final estimation is made with a global refinement step, using bundle adjustment to minimize the epipolar error (or the re-projection error in the case of 6-point) while also making the intrinsic parameters across the different frames converge to constant values. This can be achieved by introducing the variance of intrinsic parameters along the trajectory as additional terms of the cost function to be minimized. Adding the variance of each intrinsic parameter is convenient because the new cost function is still a squared sum of residuals, allowing the use of the same least squares techniques from standard bundle adjustment. In this step we assume a camera with zero-skew and four parameters to converge

$$\mathsf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} . \tag{5.21}$$

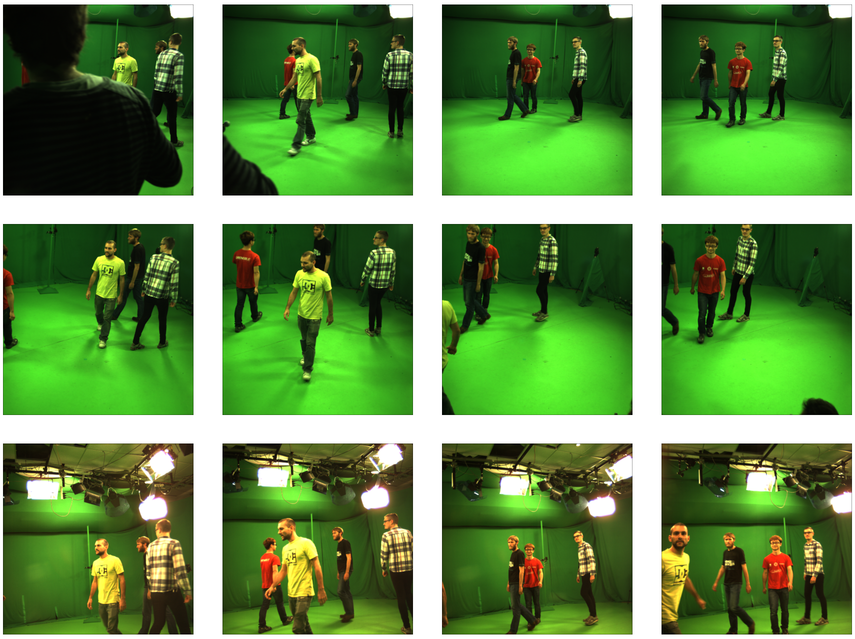
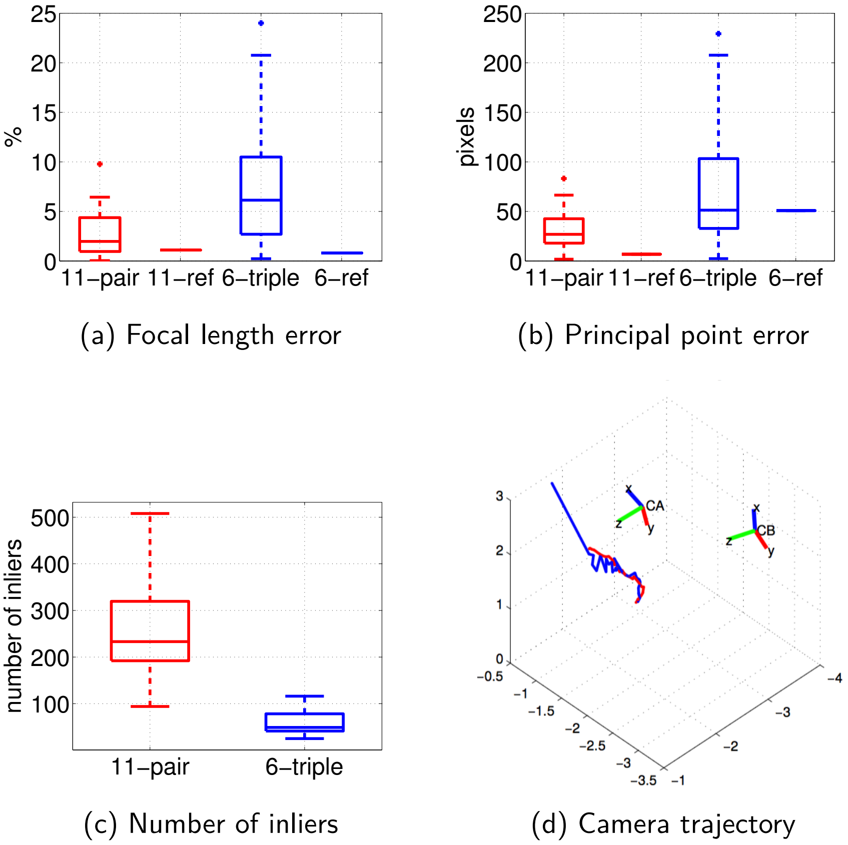


Figure 5.7: Sample frames from the fixed calibrated cameras (top and center rows) and the uncalibrated hand-held camera (bottom row).

The error distribution for the intrinsic parameters before and after global refinement is presented in Figs. 5.8(a) and 5.8(b). The initialization results are in line with the previous experiment, with 11-pairwise providing more accurate results than the 6-triplets. These initialization results are sufficient for the focal length to converge to similarly accurate values with both algorithms, however, our algorithm is able to provide a much better estimation of the principal point due to its significantly higher number of inlier correspondences (Fig. 5.8(c)). This is also reflected in the estimation of the camera trajectory (Fig. 5.8(d)). The trajectory estimated by our algorithm seems more likely to represent a hand-held trajectory, since the estimation from 6-triplets contains some non-smooth spikes. Since we do not have groundtruth values for the camera trajectory, in Fig. 5.7.3 both trajectories are projected onto the image plane of a third calibrated camera in which the person handling the free camera is visible. This figure contains 3 super-imposed images corresponding to 3 different frames. We can compare the accuracy of 11-pairwise and 6-triplets by checking if the projected trajectory coincides with the position of the hand-held camera as captured in the images. This confirms the intuition from Fig. 5.8(d) that our algorithm provides better trajectory estimations.



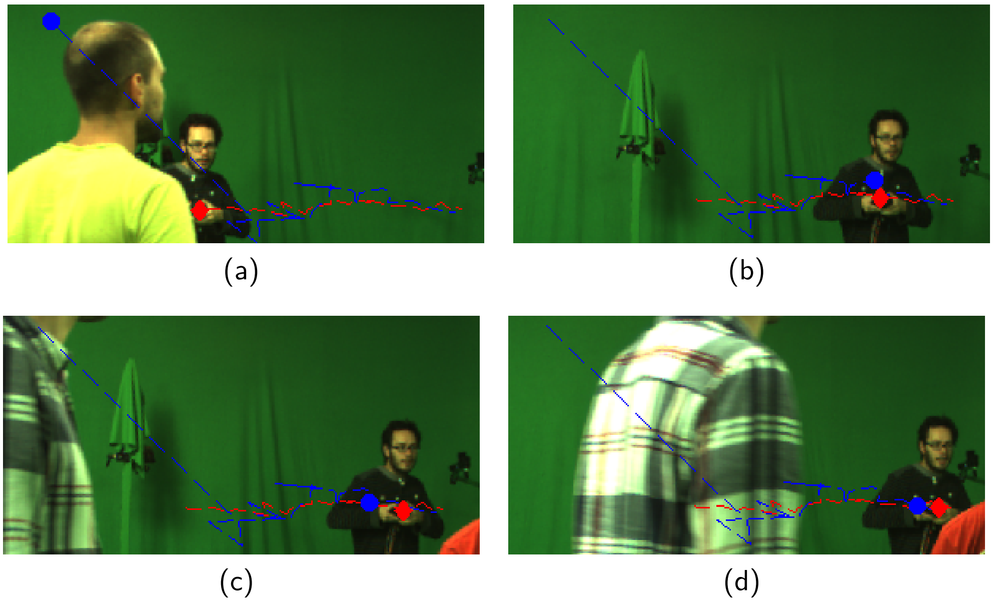
(a) Focal length error

(b) Principal point error

(c) Number of inliers

(d) Camera trajectory

Figure 5.8: Results for the hand-held camera calibration with the 11-point (red) and the 6-point (blue) algo- rithms: (a), (b) Error distributions for the estimated intrinsic parameters across the sequence before (11- pair, 6-triple) and after global refinement (11-ref, 6-ref). (c) Distribution of the number of inliers. (d) Camera trajectory.



(a)

(b)

(c)

(d)

Figure 5.9: Hand-held camera trajectory estimated by 11-point (red) and 6-point (blue) as viewed by an additional stationary camera. The points of the trajectory corresponding to each frame are shown with markers.

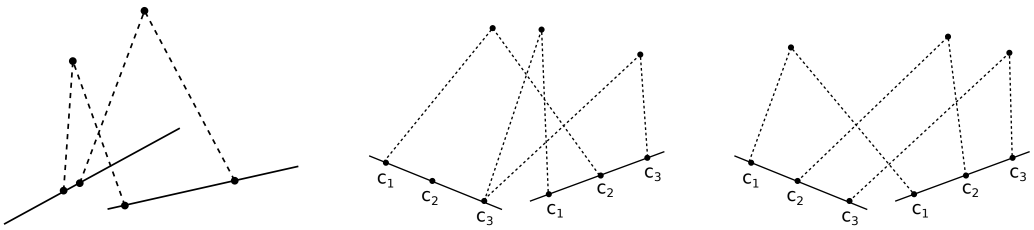
Chapter 6

Relative pose between Axial cameras

6.1 Introduction

Vision systems can be classified as being central or non-central [91]. A particular imaging device is central iff all the back-projection rays intersect in a single point in 3D, i. e., the viewpoint of the camera. Whenever a vision system has more than one viewpoint it is said to be non-central. An axial camera is a particular case of a non-central camera where every back-projection ray intersects a line in 3D (the axis). The axial camera can be used to model vision systems and imaging situations of practical interest. Examples include any catadioptric system that combines a revolution mirror with a central camera for which the viewpoint is aligned with the mirror axis (e.g. a pinhole looking at a spherical mirror) [91]; the situation of a perspective camera looking through multiple flat refractive mediums [92]; or a multi-camera rig composed by two or more pinhole cameras with collinear optical centers [6].

This chapter addresses the problem of estimating the relative pose between two axial cameras using point correspondences. Pless showed that the Plücker coordinates of two corresponding back-projection rays must satisfy a bilinear constraint that can be expressed by a 6x6 matrix [18]. This general essential matrix can be estimated from a minimum of 17 point correspondences using a DLT like approach, and its result factorized into relative camera rotation and translation. Later in [93] Sturm observed that for the case of axial cameras the estimation of this 6x6 matrix was under-determined. He proposed a new 5x5 essential matrix that can be linearly



(a) Generic axial camera (16-point algorithm)

(b) Multi-camera rig with cross-correspondences (16-point algorithm)

(c) Multi-camera rig with correspondences between the same camera (14-point algorithm)

Figure 6.1: Axial camera configurations

determined from 16 point correspondences. More recently Kim *et al.* investigated the problem of motion estimation using a camera rig composed by multiple perspective cameras with aligned optical centers [6]. They confirmed Sturm's result for the case of considering cross-correspondences between different cameras in the rig (Fig. 6.1(b)). However, if only matches between the same cameras are allowed, then it is possible to linearly determine the relative motion using a minimum of 14 point correspondences (Fig. 6.1(c)).

Neither the 16-point solution described by Sturm [93], which is applicable to any axial camera, nor the 14-point algorithm proposed in [6], that is specific to non-overlapping multiple camera rigs, are minimal solutions. The relative pose problem has 6 unknowns meaning that in theory 6 point correspondences provide enough information for determining the relative rotation and translation of the axial camera. Stewenius et al. proposed in [16] a minimal solution for the relative pose between generalized cameras. However, their algorithm is complex, provides a large number of possible solutions (up to 64), and, as reported in [6], it degenerates for most axial camera configurations. This article does not provide a minimal solution for the relative pose between axial cameras, but shows how the motion can be computed using as few as 10 point correspondences. Our 10-point method is an advance with respect to the previous 16-point [6,93] and 14-point [6] algorithms, that improves the accuracy and efficiency of motion estimation using hypothesize-and-test frameworks.

Please note that, although the 10-point algorithm generalizes to any axial camera, most of the derivations and experiments have in mind the particular case of a conventional stereo camera rig. There are prior works proposing minimal solutions for 6D stereo visual odometry but they either consider sets of features observed, respectively, in four, three, and two views [20], or use five point correspondences between two particular views plus a sixth match for solving for the scale [94]. Unlike these works, we model the stereo rig as a generic axial camera and make no assumptions about the matches. Since any pairwise correspondence can be used as input, the sampling of the solution space is more thorough, being possible to obtain correct motion estimation in circumstances for which the methods of [20, 94] are unable to provide a solution.

6.2 A new parametrization for axial cameras

Sturm describes in [93] a 5x5 essential matrix that relates back-projection rays of two axial views. We provide a different parametrization of this matrix that, not only enables to understand the results described in [6], but also proves to be useful in deriving polynomial equations that will constraint the motion estimation.

6.2.1 Linear subspace for back-projection rays

Our parameterization for axial cameras directly extends the formulation derived in previous chapter, in which we use a 5D linear subspace that represents all rays from two calibrated cameras.

We define two reference frames O and $\widehat{\mathsf{O}}$ along the camera axis $\mathbf{B}$ as depicted in Fig. 6.2(a), with an arbitrary baseline b. The transformation of homogeneous coordinates from O to $\widehat{\mathsf{O}}$ is given by a reflection

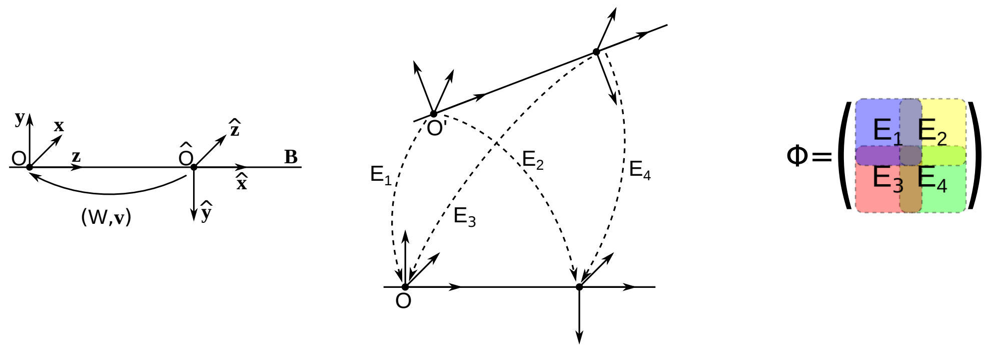
$$\mathsf{W} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \tag{6.1}$$

and a translation

$$\mathbf{v} = \begin{pmatrix} 0 & 0 & b \end{pmatrix}^{\mathsf{T}} \tag{6.2}$$

Given that all back-projection rays $\mathbf{L}_i$ of an axial camera intersect the axis $\mathbf{B}$, they belong to a linear line congruent of dimension 4 [86]. This means that all rays can be represented as a linear combination of 5 base lines $\mathbf{L}_x$, $\mathbf{L}_y$, $\mathbf{L}_z$, $\widehat{\mathbf{L}}_y$, $\widehat{\mathbf{L}}_z$. Therefore there is a vector λ_i such that for any line $\mathbf{L}_i$ the following holds

$$\mathbf{L}_i = \underbrace{\begin{pmatrix} \mathbf{L}_x & \mathbf{L}_y & \mathbf{L}_z & \widehat{\mathbf{L}}_y & \widehat{\mathbf{L}}_z \end{pmatrix}}_{\Gamma} \lambda_i \tag{6.3}$$



(a) Representation of reference frames along the camera axis

(b) The four 3×3 essential matrices encoded by Φ

(c) 5×5 essential matrix for axial cameras

Figure 6.2: A new parameterization for axial cameras

These 5 lines that compose the linear mapping $\mathsf{\Gamma}$ can be arbitrarily chosen, assuring that they intersect $\mathbf{B}$ and are linearly independent. For the purpose of our formulation we align these lines respectively with the axes $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$, $\widehat{\mathbf{y}}$, $\widehat{\mathbf{z}}$ in Fig. 6.2(a) and therefore

$$\mathsf{\Gamma} = \begin{pmatrix} \mathsf{I}_{3\times3} & \mathsf{W}^{\{1:3,2:3\}} \\ 0_{3\times3} & [\mathbf{v}]_\times^{\{1:3,2:3\}} \end{pmatrix} \tag{6.4}$$

6.2.2 Essential matrix for axial cameras

Given a set of intersecting ray correspondences $(\mathbf{L}_i, \mathbf{L}'_i)$, we can establish linear relations using the generalized camera model introduced by Pless [18]

$$\mathbf{L}_i{}^{\mathsf{T}} \begin{pmatrix} [\mathbf{t}]_\times \mathsf{R} & \mathsf{R} \\ \mathsf{R} & 0 \end{pmatrix} \mathbf{L}'_i = 0 \tag{6.5}$$

Taking into account equation 6.3 this expression can be rewritten as

$$\lambda_i{}^{\mathsf{T}} \underbrace{\mathsf{\Gamma}^{\mathsf{T}} \begin{pmatrix} [\mathbf{t}]_\times \mathsf{R} & \mathsf{R} \\ \mathsf{R} & 0 \end{pmatrix} \mathsf{\Gamma}}_{\Phi} \lambda'_i = 0 \tag{6.6}$$

with $\mathsf{\Phi}$ being the 5×5 essential matrix for axial cameras. By substituting equation 6.4 into $\mathsf{\Phi}$ we can define the following matrices

$$\mathsf{E}_1 = \mathsf{\Phi}^{\{1:3,1:3\}} = [\mathbf{t}]_\times \mathsf{R} \tag{6.7}$$

$$\mathsf{E}_2 = \mathsf{\Phi}^{\{1:3,3:5\}} = [\mathsf{R}\mathbf{v} + \mathbf{t}]_\times \mathsf{R}\mathsf{W} \tag{6.8}$$

$$\mathsf{E}_3 = \mathsf{\Phi}^{\{3:5,1:3\}} = [\mathsf{W}^\mathsf{T}(\mathbf{t} - \mathbf{v})]_\times \mathsf{W}^\mathsf{T}\mathsf{R} \tag{6.9}$$

$$\mathsf{E}_4 = \mathsf{\Phi}^{\{3:5,3:5\}} = [\mathsf{W}^\mathsf{T}(\mathsf{R}\mathbf{v} + \mathbf{t} - \mathbf{v})]_\times \mathsf{W}^\mathsf{T}\mathsf{R}\mathsf{W} \tag{6.10}$$

The placement of E_1, E_2, E_3, E_4 within $\mathsf{\Phi}$ can be better visualized in Fig. 6.2(c). From the above expressions it can be observed that E_1, E_2, E_3, E_4 are the 3×3 essential matrices that encode the motions represented in Fig. 6.2(b).

From the equations 6.7 to 6.10 we can also derive the following relations

$$\mathsf{E}_1 - \mathsf{E}_2\mathsf{W}^\mathsf{T} - \mathsf{W}\mathsf{E}_3 + \mathsf{W}\mathsf{E}_4\mathsf{W}^\mathsf{T} = 0 \tag{6.11}$$

$$\mathsf{E}_1[\mathbf{v}]_\times + [\mathbf{v}]_\times\mathsf{E}_1 - [\mathbf{v}]_\times\mathsf{E}_2\mathsf{W}^\mathsf{T} - \mathsf{W}\mathsf{E}_3[\mathbf{v}]_\times = 0 \tag{6.12}$$

These constraints provide 8 linear equations on the parameters of $\mathsf{\Phi}$ and therefore they can be used to reduce this matrix from its 25 parameters to a linear combination of 17 parameters using Gaussian elimination. This means that it is possible to estimate $\mathsf{\Phi}$ with a DLT like approach using 16 correspondences, which is in conformity with the linear formulations introduced in [93].

6.2.3 A particular axial camera: the stereo rig

The above formulation applies directly to the case of motion estimation between stereo pairs, however, there are particular configurations that require additional considerations. Furthermore, in this case it is advantageous to consider that reference frames O and $\widehat{\mathsf{O}}$ from Fig. 6.2(a) are coincident with the principal points of the stereo pair, with the translation $\mathbf{v}$ being the baseline between the cameras. This way all line coordinates in the left and right cameras will have the form

$$\lambda_{left} = \begin{pmatrix} l_1 & l_2 & l_3 & 0 & 0 \end{pmatrix}^\mathsf{T} \tag{6.13}$$

$$\lambda_{right} = \begin{pmatrix} 0 & 0 & l_3 & l_4 & l_5 \end{pmatrix}^\mathsf{T} \tag{6.14}$$

Considering the 4 cameras in this scenario, $\{\mathsf{C}_{left}, \mathsf{C}_{right}, \mathsf{C}'_{left}, \mathsf{C}'_{right}\}$, there are 4 different types of correspondences that can be used: $(\mathsf{C}_{left}, \mathsf{C}'_{left})$, $(\mathsf{C}_{left}, \mathsf{C}'_{right})$, $(\mathsf{C}_{right}, \mathsf{C}'_{left})$, $(\mathsf{C}_{right}, \mathsf{C}'_{right})$. If at least 3 of these types of correspondences are

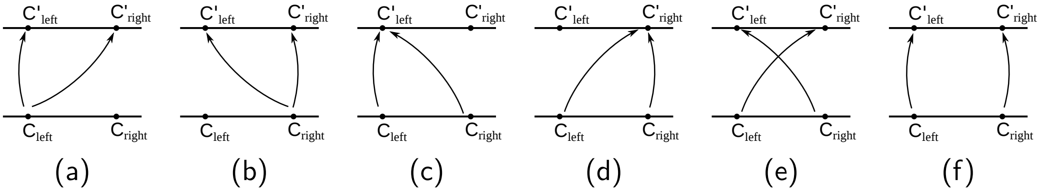


Figure 6.3: Different configurations when only two types of correspondences are established between two stereo pairs.

available then we can use the formulation from section 6.2.2. However, when only 2 types are available Φ cannot be fully known using just linear constraints and thus we further reduce the number of parameters to estimate. The cases depicted in Fig. 6.3(a) to 6.3(d) only have linear constraints on two contiguous essential matrices that share 3 parameters (see Fig. 6.2(c)), and therefore we can reduce Φ to either a 3×5 or a 5×3 matrix, which can be linearly estimated from just 14 correspondences. The cases depicted in Fig. 6.3(e) and 6.3(f) only have linear constraints on essential matrices that share one single element (either E_1 and E_4 or E_2 and E_3), resulting in 17 parameters. However, we must note that the following relations

$$\mathsf{F}_1 = \mathsf{W}\mathsf{E}_4\mathsf{W}^\mathsf{T} - \mathsf{E}_1 = \mathsf{R}[\mathbf{v}]_\times - [\mathbf{v}]_\times\mathsf{R} \tag{6.15}$$

$$\mathsf{F}_2 = \mathsf{E}_2\mathsf{W}^\mathsf{T} - \mathsf{W}\mathsf{E}_3 = \mathsf{R}[\mathbf{v}]_\times + [\mathbf{v}]_\times\mathsf{R} \tag{6.16}$$

imply that the diagonals of F_1 and F_2 are respectively $\begin{pmatrix} a & -a & 0\end{pmatrix}^\mathsf{T}$ and $\begin{pmatrix} a & a & 0\end{pmatrix}^\mathsf{T}$, with a being an unkown scalar value. In either case these equations provide 2 linear constraints that enable to reduce the number of parameters to 15, and therefore we are able to compute them from 14 correspondences.

It is noteworthy that [6] only addresses the case depicted in Fig. 6.3(f), and that in the previous chapter we address the cases 6.3(a) to 6.3(d). On the other hand our analysis in this section covers all the cases.

6.3 Towards a minimal solution

Henceforth we will only address the general axial case, but analogous conclusions can be drawn for the particular cases of Fig. 6.3.

Using equations 6.7 to 6.10 we can write the following expression

$$\alpha \mathsf{E}_1 + \beta \mathsf{E}_2 \mathsf{W}^\mathsf{T} + \gamma \mathsf{W}\mathsf{E}_3 + \delta \mathsf{W}\mathsf{E}_4\mathsf{W}^\mathsf{T} = [\alpha \mathbf{t} + \beta(\mathsf{R}\mathbf{v} + \mathbf{t}) + \gamma(\mathbf{t} - \mathbf{v}) + \delta(\mathsf{R}\mathbf{t} + \mathbf{t} - \mathbf{v})]_\times \mathsf{R} \tag{6.17}$$

with α, β, γ, δ being any real values. This means that any matrix E_i that is a linear combination of E_1, $\mathsf{E}_2\mathsf{W}^\mathsf{T}$, $\mathsf{W}\mathsf{E}_3$, $\mathsf{W}^\mathsf{T}\mathsf{E}_4\mathsf{W}$ is itself an essential matrix, verifying the cubic constraints

$$2\mathsf{E}_i{\mathsf{E}_i}^\mathsf{T}\mathsf{E}_i - tr(\mathsf{E}_i{\mathsf{E}_i}^\mathsf{T})\mathsf{E}_i = 0 \tag{6.18}$$

$$\det \mathsf{E}_i = 0 \tag{6.19}$$

From this result we are able to generate a high amount of polynomial equations by choosing different values for α, β, γ, δ. Using simulated data we were able to find 78 linearly independent equations. These equations can then be used in a similar fashion to [7] in order to reduce the number of required correspondences to solve our problem. If we use $16 - N$ correspondences, we can compute a $N + 1$ dimensional linear subspace using the equations 6.6, 6.11 and 6.12. Posteriorly we introduce this subspace into instances of equations 6.18 and 6.19 to form a polynomial system in N variables.

The minimal solution for this problem requires only 6 correspondences, which means that a polynomial system in 10 variables would need to be solved. However, as the number of variables grows, the more difficult it becomes to generate a numerically stable polynomial solver. Solving polynomial systems can be achieved with the action matrix method described in [10].

Solving a polynomial system using the action matrix method requires a minimum number of linearly independent polynomial equations, which is usually determined on a case by case basis. However, it is a general rule of thumb that this number increases with the number of different monomials that are present in the equations. If for a given system we have less equations than the minimum required, it means that new higher order equations need to be generated by multiplying the existing ones by other polynomials, resulting in new equations that although being redundant can be useful to solve the system if they are linearly independent.

Equations 6.18 and 6.19 always produce dense cubic polynomials, which means that they generally have non-zero coefficients for all monomials up to the 3rd degree. In this particular case, the system is guaranteed to be solvable if the number of

n. of variables	1	2	3	4	5	6	7	8	9	10
n. of monomials	4	10	20	35	56	84	120	165	220	286
n. of leading monomials	1	4	10	20	35	56	84	120	165	220

Table 6.1: Number of monomials in dense cubic polynomials

linearly independent equations is greater than or equal to the number of leading monomials (3rd degree). Note however that this is not a necessary condition.

In table 6.1 we list the number of leading monomials for a varying number of variables in a dense cubic equation. Since we can generate a maximum of 78 linearly independent cubic equations, it is possible to solve the system for a maximum of 6 variables without the need for generating higher order equations. With a higher number of variables the polynomial system becomes infeasibly complex for our action matrix approach. This means that we are able to implement a 10-point algorithm using this technique, which requires 56 equations and outputs 56 possible solutions. In our implementation we generated equations from the constraints of the following set of essential matrices: $\{\mathsf{E}_1, \mathsf{E}_2\mathsf{W}^\mathsf{T}, \mathsf{W}\mathsf{E}_3, \mathsf{W}^\mathsf{T}\mathsf{E}_4\mathsf{W}, \mathsf{E}_1 + \mathsf{E}_2\mathsf{W}^\mathsf{T}, \mathsf{E}_1 + \mathsf{W}\mathsf{E}_3\}$. Note that since this is a non-minimal solution the polynomial system is over-constrained and therefore it is not possible to find its exact solution. This means that all 56 solutions determined with the action matrix method will have non-zero residue in the polynomial equations and therefore we must select the one with the minimum residue.

6.4 Additional non-linear constraints

From the analysis of equations 6.7 to 6.10 it is possible to derive the following additional quadratic constraints:

$$(\mathsf{E}_1 - \mathsf{W}\mathsf{E}_3)(\mathsf{E}_1 - \mathsf{W}\mathsf{E}_3)^\mathsf{T} = -[\mathbf{v}]_\times^2 \tag{6.20}$$

$$(\mathsf{E}_1 - \mathsf{E}_2\mathsf{W}^\mathsf{T})^\mathsf{T}(\mathsf{E}_1 - \mathsf{E}_2\mathsf{W}^\mathsf{T}) = -[\mathbf{v}]_\times^2 \tag{6.21}$$

From the analysis of the previous section it is evident that these equations can provide a great help in reducing the number of required correspondences. Since these constraints are quadratic, while the constraints discussed in the previous section are cubic, it is possible to multiply equations 6.20 and 6.21 by each of the unknown variables in the system to obtain additional polynomial equations without increasing the number of existent monomials. Since we can use this method to generate more than 6 linearly independent quadratic and cubic equations, we can build a polynomial system with more than 84 equations and thus, by analysis of table 6.1, we can remove at least one more variable from the non-linear system and solve the problem from 9 correspondences.

In this thesis we do not pursue this analysis further, however, it still remains unclear whether these equations can be used to simplify the polynomial system, e.g. by Gaussian elimination of some monomials. These observations call for a more thorough investigation of this problem in future work.

6.5 The 10-point algorithm

In this section we summarize the steps required to estimate the relative pose (R, $\mathbf{t}$) between two calibrated axial cameras using pairwise point correspondences. Each point from an axial camera is represented by its image ray as line coordinates $\mathbf{L}_i$. Consider as inputs 10 or more correspondences ($\mathbf{L}_i$, $\mathbf{L}'_i$) between the two axial systems.

1. Map correspondences $(\mathbf{L}_i, \mathbf{L}'_i)$ into 4D homogeneous coordinates (λ_i, λ'_i) taking into account the relation of equation 6.3.

2. Stack 10 instances of equation 6.6 in terms of the 25 parameters of Φ, and then use equations 6.11 and 6.12 to eliminate 8 parameters in Φ

3. Generate a 7 dimensional linear subspace of solutions using SVD. Since the linear solution is up to scale, one of the parameters can be set to 1, resulting in 6 unkowns.

4. Introduce the linear expression with 6 unknowns into the polynomial constraints resulting from the following set of essential matrices: $\{\mathsf{E}_1, \mathsf{E}_2\mathsf{W}^\mathsf{T}, \mathsf{W}\mathsf{E}_3, \mathsf{W}^\mathsf{T}\mathsf{E}_4\mathsf{W}, \mathsf{E}_1 + \mathsf{E}_2\mathsf{W}^\mathsf{T}, \mathsf{E}_1 + \mathsf{W}\mathsf{E}_3\}$.

5. Use 56 polynomial equations to compute the action matrix, and obtain 56 solutions that correspond to its eigenvectors.

6. Substitute all 56 solutions back into the polynomial system and select the solution with the smallest residue as the correct one.

7. Project E_1, E_2, E_3, E_4 individually onto the essential matrix manifold using SVD decomposition.

8. For each of the four 3×3 essential matrices, make an independent factorization to find R and estimate $\mathbf{t}$ with the correct scale by substituting R into equation 6.6.

9. From the four different estimations of $(\mathsf{R}, \mathbf{t})$ choose the one with minimum residue in equation 6.6 (alternatively, measure the re-projection errors).

6.6 Experimental Validation

In this section we validate our algorithm with both synthetic and real data in the case of pose estimation between stereo cameras. We allow all types of correspondences and therefore we use the 17-parameter formulation of the problem.

We test our algorithm using minimal data in the first synthetic experiment and then compare our algorithm against the 16-point linear approach from Kim *et al.* [6] with both synthetic and real data. In its original version [6] has a refinement step that alternates between successive translation and rotation estimations. We do not implement this step in our experiments because it aims at being fast and simple, sacrificing some robustness and optimality. This step would not likely be used in a real scenario, where refinement would be accomplished with bundle adjustment. Additionally, we compare both algorithms within RANSAC.

Please note that, unlike other stereo relative pose algorithms [20,29], both the 16-point and our 10-point work with arbitrary distributions of correspondences across the different image pairs. This means that our multiset-RANSAC approach is not required in this context and instead we use a standard RANSAC where correspondences across all pairs of cameras are sampled indiscriminately.

6.6.1 Synthetic data

We built a simulated environment in which two stereo camera pairs are randomly positioned with overlapping field of views, and in front of them a set of 3D points is randomly generated within a bounded region. The input to the algorithms are the back-projections of the 3D points according to the pinhole model affected by image Gaussian noise.

Given a motion estimation $(\mathsf{R}, \mathbf{t})$ and the groundtruth values $(\mathsf{R}_{GT}, \mathbf{t}_{GT})$, the error in rotation is measured by the Euler angle of the residual rotation $\mathsf{R}^{\mathsf{T}}\mathsf{R}_{GT}$, the error in translation direction is measured as the angle defined by $\mathbf{t}$ and $\mathbf{t}_{GT}$, and

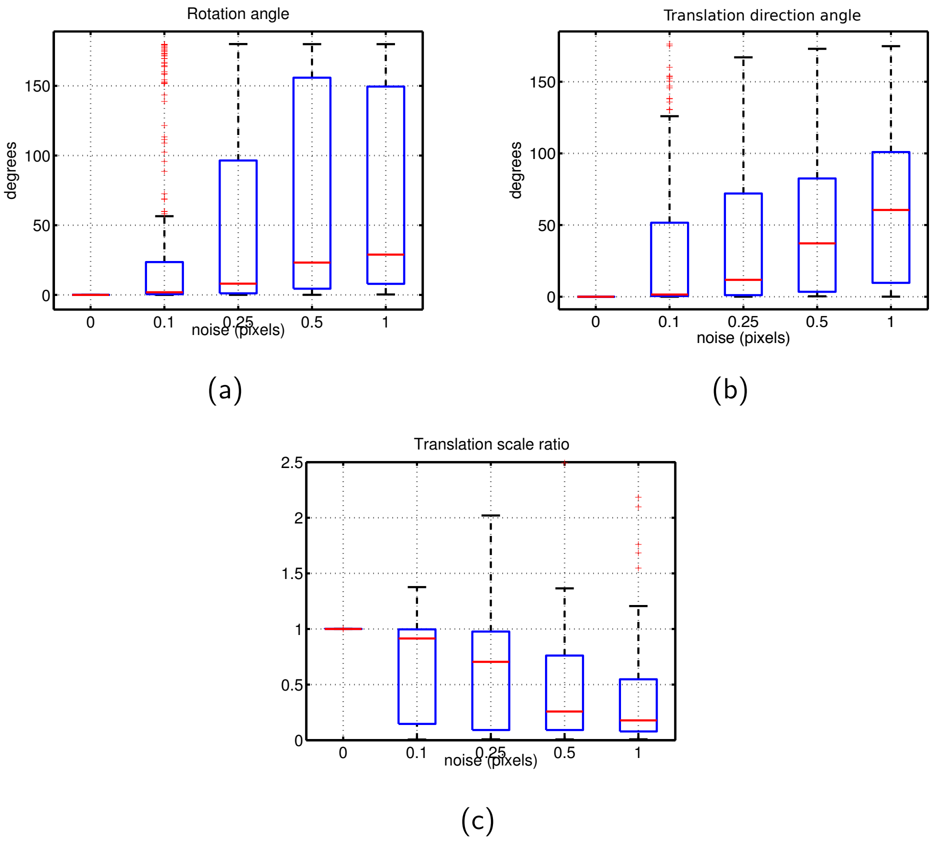
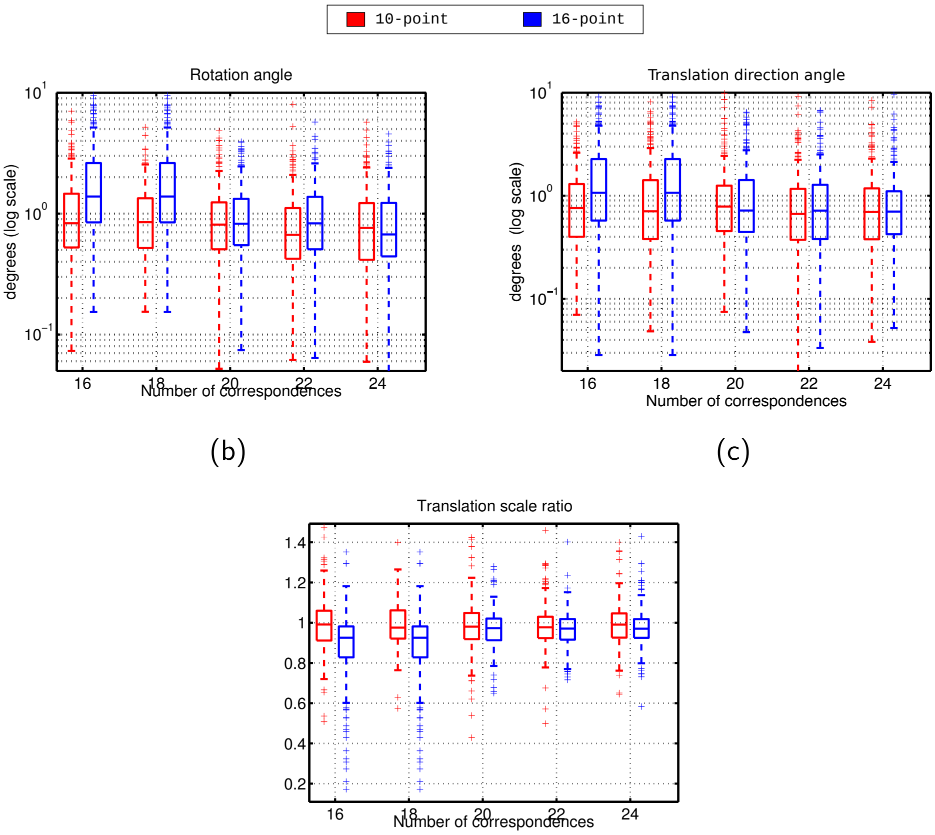


Figure 6.4: Error distributions for 10-point estimations with synthetic data. We perform 250 trials for each noise level.

the ratio $||\mathbf{t}||/||\mathbf{t}_{gt}||$ evaluates the quality in the estimation of the translation scale factor.

In a first simulation our algorithm is tested 250 times using 10 input pairwise correspondences for different noise magnitudes without RANSAC. The error distributions for translation and rotation displayed in Fig 6.4 show that in a noise free scenario our algorithm outputs the exact solution. However, for a noise magnitude over 0.5 pixels the stability decreases significantly, which suggests that a robust estimation is required to improve performance. It is also noticeable that with high levels of noise there is a bias that systematically underestimates the translation scale factor, which calls for further study of the problem in the future.

In a second simulation our algorithm is compared against the 16-point linear algorithm for a different number of input correspondences while injecting noise with 1 pixel of standard deviation. Again, 250 trials were tested for each case and the



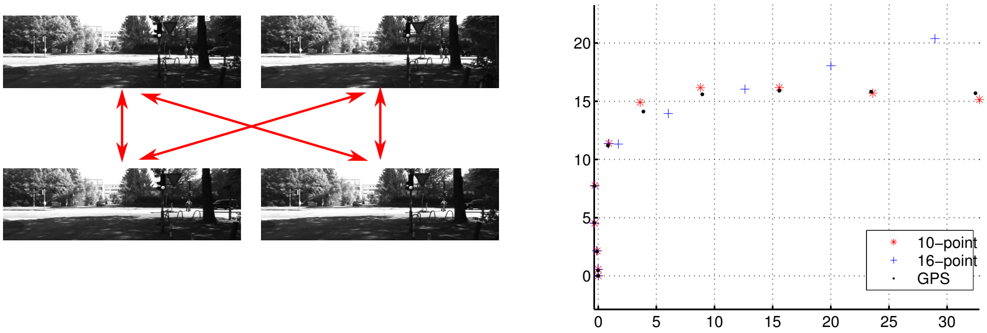
(b)

(c)

(d)

Figure 6.5: Error distributions for estimations with the 10-point and the 16-point algorithms. We perform 250 trials for number of correspondences.

error distributions for translation and rotation are displayed in Fig. 6.5. The better performance of our algorithm is specially evident when using a low number of correspondences. But it also must be noticed that for a higher number of correspondences, while the stability between both algorithms is similar, our algorithm is significantly faster due to the fact that RANSAC is sampling 10 points instead of 16. The RANSAC procedure eliminates the previously observed bias in the estimation of translation scale factor induced by our 10-point algorithm, furthermore, this effect is still visible on the 16-point algorithm for estimations with 16 and 18 correspondences, in which RANSAC provides little to none advantage.



(a) Input data

(b) Trajectory (meters)

Figure 6.6: Performance comparison between 10-point algorithm and 16-point algorithm with real data.

6.6.2 Real data

We used data from the KITTI Vision Benchmark Suite [30], which contains fully calibrated stereo sequences and GPS measurements acquired by a vehicle in an urban environment. We selected a set of images from the sequence ”2011_09_28_drive_0001” and compare the trajectories estimated by both algorithms.

Correspondences were obtained by matching SIFT features [2] on the four combinations of image pairs of consecutive frames (Fig. 6.6(a)). Both our 10-point algorithm and the 16-point algorithm are used within a RANSAC framework. Given the high amount of sampling points for both algorithms the number of RANSAC iterations can become infeasibly high. We also perform the pre-filtering step explained in Section 5.6.2, Chapter 5 by using a 5-point RANSAC [7] in all four combinations of matched image pairs. The set of input correspondences after pre-filtering contains a very high inlier ratio which significantly decreases the number of iterations required to find an accurate estimation. We use these algorithms just to make odometry estimations, i. e., the trajectory is not refined by bundle adjustment. This way the accumulation of drift error serves as an evaluation of comparative accuracy, showing that our algorithm provides more robust estimations (Fig. 6.6(b)).

Chapter 7

Conclusions

In this thesis we discuss minimal problems in different multi-camera or multi-sensor applications, namely:

- extrinsic calibration between a laser rangefinder and a camera;
- full calibration of an ultrasound array with a camera;
- full calibration of a camera within a network;
- relative pose between axial cameras.

In each of these cases our algorithms use less correspondences than the corresponding state-of-the-art alternatives. In practice, this introduces two advantages: the number of RANSAC iterations is minimized; the number of acquisitions to obtain a pre-determined level of accuracy is decreased. The first advantage is more relevant in the applications that might require real-time performance. This includes the hand-held camera experiment from Chapter 5 and the relative motion between axial cameras of Chapter 6. Note, however, that even in offline calibration procedures it might be desirable to reduce its computational time, specially in problems that deal with extremely high number of RANSAC samples (e.g. Chapter 6) where the decrease in terms of execution time from a non-minimal solution to a minimal solution might be from several hours to a few minutes. The second advantage is evident in the experimental section of the LRF-camera calibration problem (Chapter 3) when comparing the performance of the minimal solution against a non-minimal solution for a varying number of acquisitions. It is always the case that a given level of accuracy can be achieved with less acquisitions using the minimal solution, making the calibration procedure easier to perform. Getting a pre-determined level of accuracy wit the minimum amount of acquisitions is an important feature in the US calibration problem. In this case the calibration must be done in the operating room by a doctor, and thus a time consuming and complex procedure must be avoided at all costs.

All the proposed solutions are broadly based on two new insights to tackle geometric problems in computer vision. In the first and second minimal problems (Chapters 3 and 4) we observe that the registration in 3D Euclidean space between planes and co-planar lines can be re-stated in terms of their point and line intersections. In this way we are able to model this type of problems as virtual pinhole cameras observing 3D points and take useful insights from the extensive literature on absolute orientation [4, 13, 49, 72]. In the last two problems (Chapters 3 and 4) we take advantage of the fact that a pair of calibrated cameras can be modelled as a single axial sensor. We derive extended fundamental and essential matrices that are able to capture the line incidence relations of all viewpoints in an unified way. These are two general insights that can be useful in the context of other problems that either involve plane-line registration or point correspondences with stereo cameras.

The multiset-RANSAC/MLESAC/MAPSAC framework is a new contribution that adapts the RANSAC formulation to more complex random sampling schemes involving multiple datasets. This situation arises frequently in problems with multiple cameras or multiple sensors when different types of correspondences must be sampled in a specific way. This framework is especially useful in the problems from Chapters 4 and 5.

The camera network problem from Section 5 requires multiset-RANSAC to sample a very high number of correspondences. This poses the question of what is the threshold when minimal-solvers and RANSAC start to be impracticable due to the complexity of the problem at hands. Although this highly depends on computational power, which will push this threshold further over time, currently RANSAC sampling of 10 or 11 correspondences with a moderate amount of outliers might translate into several hours of runtime execution, if pre-filtering techniques are not used. In problems with a similar or higher complexity than this problem, it might be a competitive approach to use non-minimal solutions with a posterior projection on the solution space.

The contributions in this thesis also point to several possible research directions in the future.

Multiset-RANSAC is a general framework that can be used in many different problems outside the scope of this thesis (e.g. [95, 96]). With some additional modifications, multiset-RANSAC can also be extended to other sampling schemes that were not considered in here, namely when different datasets cannot be selected indiscriminately [20], and when there is not a strict number of samples to select from each dataset (the stereo rig application in Chapter 6). Some more intelligent sampling methods, e.g. PROSAC [97] or guided-MLESAC [98], could also be adapted to the multiple-dataset framework to improve the computational efficiency of multiset-RANSAC. This could be an alternative to the pre-filtering step used in the camera network and the axial camera problems.

Multiset-RANSAC can also be used in the LRF-Camera calibration problem to automatically detect lines in the LRF-Camera calibration procedure, by sampling directly the LRF depth measurements instead of previously estimated lines. This, however, would drastically increase the outlier ratio in the input data (all depth measurements that do not belong to the grid) and make the use of a minimal 3-correspondence solution significantly more crucial.

The US-Camera calibration method still needs to be validated more thoroughly in practice and extended to applications that are closer to the real medical environment. The accuracy of line detection can be improved using a Cambridge phantom or a thin membrane. This method should also be tested on close-range US probes (7.5 – 10 MHz) that would likely provide better line measurements with less reflections. The accuracy of this method should also be validated with data from real medical imagery, e.g., to guide an instrumented tool through the US image.

The camera calibration within a network can in principle be solved minimally with either 7 and 4 pairwise correspondences from different cameras, or alternatively 6 and 5 correspondences. This second case has not been solved yet, as the resulting non-linear constraints is harder to solve. It also remains unanswered to which extent an 11 point algorithm is more efficient and/or reliable than just performing independent 7-point estimations (pre-filtering step) and stepping directly to the non-linear iterative refinement, avoiding the multiset-RANSAC altogether.

Finally, the minimal solution to the relative pose between axial systems is still an unsolved problem. Section 6.4 briefly mentions a set of additional polynomial equations that are not used in the 10-point algorithm proposed in this thesis, which calls for further investigation of this problem.

Bibliography

[1] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[2] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vision*, vol. 60, pp. 91–110, November 2004.

[3] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision.* Cambridge Academic Press, 2003.

[4] R. M. Haralick, C.-N. Lee, K. Ottenberg, and M. Nölle, “Review and analysis of solutions of the three point perspective pose estimation problem,” *International Journal of Computer Vision*, vol. 13, pp. 331–356, December 1994.

[5] H. C. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections,” *Nature*, vol. 293, p. 133–135, 1987.

[6] J.-H. Kim, H. Li, and R. Hartley, “Motion estimation for nonoverlapping multicamera rigs: Linear algebraic and L_∞ geometric solutions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, pp. 1044–1059, June 2010.

[7] D. Nistér, “An efficient solution to the five-point relative pose problem,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 756–777, June 2004.

[8] H. Li and R. Hartley, “Five-point motion estimation made easy,” in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 1, pp. 630–633, 2006.

[9] H. Stewenius, *Grö bner Basis Methods for Minimal Problems in Computer Vision.* PhD thesis, Lund University, 2005.

[10] M. Byröd, K. Josephson, and K. Åström, “Fast and stable polynomial equation solving and its application to computer vision,” *Int. J. Comput. Vision*, vol. 84, pp. 237–256, Sept. 2009.

[11] Z. Kukelova, M. Bujnak, and T. Pajdla, “Automatic generator of minimal problem solvers,” in *Computer Vision – ECCV 2008* (D. Forsyth, P. Torr, and A. Zisserman, eds.), vol. 5304 of *Lecture Notes in Computer Science*, pp. 302–315, Springer Berlin Heidelberg, 2008.

[12] “Minimal problems in computer vision.” http://cmp.felk.cvut.cz/mini/.

[13] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, “Closed-form solution of absolute orientation using orthonormal matrices,” *Journal of the Optical Society of America A*, vol. 5, pp. 1127–1135, Jul 1988.

[14] K. Josephson and M. Byrod, “Pose estimation with radial distortion and unknown focal length,” in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2419–2426, June 2009.

[15] Z. Kukelova, M. Byröd, K. Josephson, T. Pajdla, and K. Åström, “Fast and robust numerical solutions to minimal problems for cameras with radial distortion,” *Computer Vision and Image Understanding*, vol. 114, no. 2, pp. 234 – 244, 2010. Special issue on Omnidirectional Vision, Camera Networks and Non-conventional Cameras.

[16] H. Stewénius, D. Nistér, M. Oskarsson, and K. Åström, “Solutions to minimal generalized relative pose problems,” in *Workshop on Omnidirectional Vision*, (Beijing China), 2005.

[17] D. Nister, “A minimal solution to the generalised 3-point pose problem,” in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, pp. I–560–I–567 Vol.1, June 2004.

[18] R. Pless, “Using many cameras as one,” in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2003.

[19] B. Clipp, C. Zach, J.-M. Frahm, and M. Pollefeys, “A new minimal solution to the relative pose of a calibrated stereo camera with small field of view overlap,” in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1725–1732, IEEE, 2009.

[20] E. Dunn, B. Clipp, and J.-M. Frahm, “A geometric solver for calibrated stereo egomotion,” in *ICCV*, 2011.

[21] F. Fraundorfer, P. Tanskanen, and M. Pollefeys, “A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles,” in *Computer Vision – ECCV 2010* (K. Daniilidis, P. Maragos, and N. Paragios, eds.), vol. 6314 of *Lecture Notes in Computer Science*, pp. 269–282, Springer Berlin Heidelberg, 2010.

[22] Z. Kukelova, M. Bujnak, and T. Pajdla, “Closed-form solutions to minimal absolute pose problems with known vertical direction,” in *Computer Vision – ACCV 2010* (R. Kimmel, R. Klette, and A. Sugimoto, eds.), vol. 6493 of *Lecture Notes in Computer Science*, pp. 216–229, Springer Berlin Heidelberg, 2010.

[23] C. Raposo, J. Barreto, and U. Nunes, “Fast and accurate calibration of a kinect sensor,” in *3D Vision - 3DV 2013, 2013 International Conference on*, pp. 342–349, June 2013.

[24] F. Vasconcelos, J. Barreto, and U. Nunes, “A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2012.

[25] F. Vasconcelos, J. P. Barreto, and E. Boyer, “A minimal solution for camera calibration using independent pairwise correspondences,” in *Computer Vision–ECCV 2012*, pp. 724–737, Springer, 2012.

[26] F. Vasconcelos and J. P. Barreto, “Towards a minimal solution for the relative pose between axial cameras,” in *BMVC*, 2013.

[27] P. H. Torr and A. Zisserman, “Mlesac: A new robust estimator with application to estimating image geometry,” *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[28] S. Choi, T. Kim, and W. Yu, “Performance evaluation of ransac family,” *Journal of Computer Vision*, vol. 24, no. 3, pp. 271–300, 1997.

[29] B. Clipp, J.-H. Kim, J.-M. Frahm, M. Pollefeys, and R. Hartley, “Robust 6dof motion estimation for non-overlapping, multi-camera systems,” in *Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on*, pp. 1–8, Jan 2008.

[30] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, (Providence, USA), June 2012.

[31] P. H. S. Torr, "Bayesian model estimation and selection for epipolar geometry and generic manifold fitting," *International Journal of Computer Vision*, vol. 50, no. 1, pp. 35–61, 2002.

[32] C. Früh and A. Zakhor, "An automated method for large-scale, ground-based city model acquisition," *International Journal of Computer Vision*, vol. 60, pp. 5–24, 2004.

[33] B. Douillard, D. Fox, and F. Ramos, "Laser and vision based outdoor object mapping," *Robotics: Science and Systems*, 2008.

[34] C. Premebida, O. Ludwig, and U. Nunes, "Lidar and vision-based pedestrian detection system," *Journal of Field Robotics*, vol. 26, pp. 696–711, September 2009.

[35] F. T. Ramos, J. Nieto, and H. F. Durrant-Whyte, "Recognising and modelling landmarks to close loops in outdoor slam.," in *IEEE International Conference on Robotics and Automation (ICRA'07)*, pp. 2036–2041, 2007.

[36] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04)*, vol. 3, pp. 2301 – 2306, 2004.

[37] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, *An invitation to 3-D vision: from images to geometric models.* Springer, 2004.

[38] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," *Vision Algorithms: Theory and Practice*, pp. 153–177, 1983.

[39] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," *IEEE International Conference on Computer Vision (ICCV'99)*, vol. 1, pp. 666–673 vol.1, 1999.

[40] J. P. Barreto, J. Roquette, P. Sturm, and F. Fonseca, "Automatic camera calibration applied to medical endoscopy," in *British Machine Vision Conference (BMVC'09)*, 2009.

[41] R. Unnikrishnan and M. Hebert, “Fast extrinsic calibration of a laser rangefinder to a camera,” tech. rep., Carnegie Mellon University, 2005.

[42] D. Scaramuzza and A. Harati, “Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes,” in *IEEE International Conference on Intelligent Robots and Systems (IROS’07)*, 2007.

[43] C. Gao and J. R. Spletzer, “On-line calibration of multiple lidars on a mobile vehicle platform,” in *IEEE International Conference on Robotics and Automation (ICRA’10)*, pp. 279–284, 2010.

[44] G. Li, Y. Liu, L. Dong, X. Cai, and D. Zhou, “An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*, pp. 3854 –3859, 29 2007-nov. 2 2007.

[45] P. Núñez, P. D. Jr, R. Rocha, and J. Dias, “Data fusion calibration for a 3d laser range finder and a camera using inertial data,” in *European Conference on Mobile Robots (ECMR’09)*, pp. 31–36, 2009.

[46] H. Aliakbarpour, P. Nuez, J. Prado, K. Khoshhal, and J. Dias, “An efficient algorithm for extrinsic calibration between a 3d laser range finder and a stereo camera for surveillance,” in *International Conference on Advanced Robotics (ICAR’09)*, pp. 1 –6, june 2009.

[47] K. D. O. Naroditsky, A. Patterson, “Automatic alignment of a camera with a line scan lidar system,” in *IEEE International Conference on Robotics and Automation (ICRA’11)*, 2011.

[48] V. Caglioti, A. Giusti, and D. Migliore, “Mutual calibration of a camera and a laser rangefinder.,” in *International Conference on Computer Vision Theory and Applications (VISAPP’08)*, pp. 33–42, 2008.

[49] V. Lepetit, F. Moreno-Noguer, and P. Fua, “Epnp: An accurate o(n) solution to the pnp problem,” *International Journal of Computer Vision*, vol. 81, pp. 155–166, February 2009.

[50] T. Moriya and H. Takeda, “Solving the rotation-estimation problem by using the perspective three-point algorithm,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR’00)*, vol. 1, pp. 766 –773 vol.1, 2000.

[51] C. Olsson, F. Kahl, and M. Oskarsson, "Branch-and-bound methods for euclidean registration problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 783 –794, May 2009.

[52] S. Ramalingam, Y. Taguchi, T. K. Marks, and O. Tuzel, "P2pi: a minimal solution for registration of 3d points to 3d planes," in *European Conference on Computer Vision (ECCV'10)*, (Berlin, Heidelberg), pp. 436–449, Springer-Verlag, 2010.

[53] I. Sutherland, "Three-dimensional data input by tablet," *Proceedings of the IEEE*, vol. 62, pp. 453–461, April 1974.

[54] W. J. Wolfe, D. Mathis, C. W. Sklair, and M. Magee, "The perspective view of three points," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 66–73, Jan. 1991.

[55] X. shan Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, pp. 930–943, Aug 2003.

[56] J. Grunert, "Das pothenotiscfie problem in erweiterter gestalt nebst über seine anwendungen in der geodäsie," *Grunerts Archiv für Mathematik und Physik*, vol. 1, pp. 238 –248, 1841.

[57] SICK AG, *LMS200/211/221/291 Laser Measurement Systems.*

[58] H. Talib, M. Peterhans, J. García, M. Styner, and M. González Ballester, "Information filtering for ultrasound-based real-time registration," *Biomedical Engineering, IEEE Transactions on*, vol. 58, pp. 531–540, March 2011.

[59] P. J. Stolka, P. Foroughi, M. Rendina, C. R. Weiss, G. D. Hager, and E. M. Boctor, "Needle guidance using handheld stereo vision and projection for ultrasound-based interventions," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014* (P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. Howe, eds.), vol. 8674 of *Lecture Notes in Computer Science*, pp. 684–691, Springer International Publishing, 2014.

[60] A. Fenster, D. B. Downey, and H. N. Cardinal, "Three-dimensional ultrasound imaging," *Physics in Medicine and Biology*, vol. 46, no. 5, p. R67, 2001.

[61] V. Kindratenko, "A survey of electromagnetic position tracker calibration techniques," *Virtual Reality*, vol. 5, no. 3, pp. 169–182, 2000.

[62] F. Lindseth, T. Langø, J. Bang, N. Hernes, and A. Toril, "Accuracy evaluation of a 3d ultrasound-based neuronavigation system," *Computer Aided Surgery*, vol. 7, no. 4, pp. 197–222, 2002.

[63] J. Welch, J. Johnson, M. Bax, R. Badr, and R. Shahidi, "A real-time freehand 3d ultrasound system for image-guided surgery," in *Ultrasonics Symposium, 2000 IEEE*, vol. 2, pp. 1601–1604 vol.2, Oct 2000.

[64] M. A. Vitrani and G. Morel, "Hand-eye self-calibration of an ultrasound image-based robotic system," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 1179–1185, Sept 2008.

[65] L. Mercier, T. Langø, F. Lindseth, and L. D. Collins, "A review of calibration techniques for freehand 3-d ultrasound systems," *Ultrasound in Medicine & Biology*, vol. 31, no. 2, pp. 143 – 165, 2005.

[66] D. F. Leotta, "An efficient calibration method for freehand 3-d ultrasound imaging systems," *Ultrasound in Medicine & Biology*, vol. 30, no. 7, pp. 999 – 1008, 2004.

[67] M. Peterhans, S. Anderegg, P. Gaillard, T. Oliveira-Santos, and S. Weber, "A fully automatic calibration framework for navigated ultrasound imaging," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pp. 1242–1245, Aug 2010.

[68] T. K. Chen, A. D. Thurston, R. E. Ellis, and P. Abolmaesumi, "A real-time freehand ultrasound calibration system with automatic accuracy feedback and control," *Ultrasound in Medicine & Biology*, vol. 35, no. 1, pp. 79 – 93, 2009.

[69] Y. Sato, M. Nakamoto, Y. Tamaki, T. Sasama, I. Sakita, Y. Nakajima, M. Monden, and S. Tamura, "Image guidance of breast cancer surgery using 3-d ultrasound images and augmented reality visualization," *Medical Imaging, IEEE Transactions on*, vol. 17, pp. 681–693, Oct 1998.

[70] R. Prager, R. Rohling, A. Gee, and L. Berman, "Rapid calibration for 3-d freehand ultrasound," *Ultrasound in Medicine & Biology*, vol. 24, no. 6, pp. 855 – 869, 1998.

[71] T. Langø, *Ultrasound guided surgery: image processing and navigation.* PhD thesis, Fakultet for informasjonsteknologi, 2000.

[72] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg, "A minimal solution to the generalized pose-and-scale problem," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 422–429, June 2014.

[73] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, pp. 11–15, Jan. 1972.

[74] H. Isack and Y. Boykov, "Energy-based geometric multi-model fitting," *International Journal of Computer Vision*, vol. 97, no. 2, pp. 123–147, 2012.

[75] J. Starck and A. Hilton, "Surface capture for performance-based animation," *Computer Graphics and Applications, IEEE*, vol. 27, no. 3, pp. 21 –31, 2007.

[76] T. Svoboda, D. Martinec, and T. Pajdla, "A convenient multicamera self-calibration for virtual environments," *Presence: Teleoper. Virtual Environ.*, vol. 14, no. 4, pp. 407–422, 2005.

[77] J. Barreto and K. Daniilidis, "Wide area multiple camera calibration and estimation of radial distortion," in *OMNIVIS'2004 - Int. Workshop in Omnidirectional vision, camera networks, and non-conventional cameras*, 2004.

[78] Z. Zhao and Y. Liu, "Practical multi-camera calibration algorithm with 1d objects for virtual environments," in *Multimedia and Expo, 2008 IEEE International Conference on*, pp. 1197 –1200, 2008.

[79] A. Zaharescu, R. Horaud, R. Ronfard, and L. Lefort, "Multiple camera calibration using robust perspective factorization," in *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pp. 504 –511, 2006.

[80] E. Shen and R. Hornsey, "Multi-camera network calibration with a non-planar target," *Sensors Journal, IEEE*, vol. 11, no. 10, pp. 2356 –2364, 2011.

[81] J. Courchay, A. Dalalyan, R. Keriven, and P. Sturm, "A global camera network calibration method with linear programming," in *Proceedings of the International Symposium on 3D Data Processing, Visualization and Transmission*, 2010.

[82] K. Josephson, M. Byrod, F. Kahl, and K. Åström, “Image-based localization using hybrid feature correspondences,” in *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, pp. 1 –8, 2007.

[83] R. Hartley and P. Sturm, “Triangulation,” *Computer Vision and Image Understanding*, 1997.

[84] P. Sturm and B. Triggs, “A factorization based algorithm for multi image projective structure and motion,” in *European Conference in Computer Vision*, 1996.

[85] S. Laveau and O. Faugeras, “3-d scene representation as a collection of images,” in *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on*, vol. 1, pp. 689–691 vol.1, Oct 1994.

[86] H. Pottmann and J. Wallner, *Computational line geometry.* Berlin: Springer Verlag, 1 ed., 2001.

[87] J. P. Barreto, *General central projection systems: Modeling, calibration and visual servoing.* PhD thesis, PhD Thesis, University of Coimbra, Coimbra, Portugal, 2004.

[88] M. I. Lourakis and R. Deriche, “Camera Self-Calibration Using the Singular Value Decomposition of the Fundamental Matrix: From Point Correspondences to 3D Measurements,” Tech. Rep. RR-3748, INRIA, Aug. 1999.

[89] J. Allard, J.-S. Franco, C. Merrier, E. Boyer, and B. Raffin, “The grimage platform: A mixed reality environment for interactions,” in *Computer Vision Systems, 2006 ICVS’06. IEEE International Conference on*, pp. 46–46, IEEE, 2006.

[90] “4d repository.” http://4drepository.inrialpes.fr/pages/home.

[91] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, and J. Barreto, “Camera models and fundamental concepts used in geometric computer vision,” *Foundations and Trends in Computer Graphics and Vision*, vol. 6, no. 1-2, pp. 1–183, 2011.

[92] A. Agrawal, S. Ramalingam, Y. Taguchi, and V. Chari, “A theory of multi-layer flat refractive geometry,” in *CVPR*, 2012.

[93] P. Sturm, “Multi-view geometry for general camera models,” in *CVPR*, 2005.

[94] T. Kazik, L. Kneip, J. Nikolic, M. Pollefeys, and R. Siegwart, “Real-time 6d stereo visual odometry with non-overlapping fields of view,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1529–1536, June 2012.

[95] C. Raposo, M. Lourenço, J. P. Barreto, and M. Antunes, “Plane-based odometry using an rgb-d camera,” in *BMVC*, 2013.

[96] C. Raposo, M. Antunes, and J. Barreto, “Piecewise-planar stereoscan:structure and motion from plane primitives,” in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8690 of *Lecture Notes in Computer Science*, pp. 48–63, Springer International Publishing, 2014.

[97] O. Chum and J. Matas, “Matching with prosac - progressive sample consensus,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 220–226 vol. 1, June 2005.

[98] B. Tordoff and D. Murray, “Guided-mlesac: faster image transform estimation by using matching priors,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, pp. 1523–1535, Oct 2005.