– CENTRUM SCIENTIARUM MATHEMATICARUM –

Towards Precise Localisation

Subsample Methods, Efficient Estimation and Merging of Maps

GABRIELLE FLOOD

Lund University
Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

Printed by Media-Tryck, Lund 2019 NORDIC SWAN ECOLABEL 3041 0903

LUND UNIVERSITY

Licentiate Theses in Mathematical Sciences 2019:1

ISBN 978-91-7895-374-5
ISSN 1404-028X
LUTFMA-2042-2019

Towards Precise Localisation

Towards Precise Localisation
Subsample Methods, Efficient Estimation and Merging of Maps

by Gabrielle Flood

LUND
UNIVERSITY

Centre for Mathematical Sciences
Lund University
Box 118
SE-221 00 Lund
Sweden

www.maths.lu.se

Licentiate Thesis in Mathematical Sciences 2019:1
ISSN: 1404-028X

ISBN: 978-91-7895-374-5 (print)
ISBN: 978-91-7895-375-2 (electronic)
LUTFMA-2042-2019

Printed in Sweden by Media-Tryck, Lund University, Lund 2019

Indeed, what is there that does not appear marvelous
when it comes to our knowledge for the first time?
How many things, too, are looked upon as quite impossible
until they have actually been effected?

Gaius Plinius Secundus

Abstract

Over the last couple of years audio and radio sensors have become cheaper and more common in our everyday life. Such sensors can be used to form a network, from which one can obtain distance measures by correlating the different received signals. One example of such distance measures is time-difference of arrival measurements (TDoA), which can be used to estimate the positions of the senders and receivers. The result is a 3D map of the environment, similar to what you get from doing structure from motion (SfM) with images. If a new sensor appears, the map can in turn be used to determine the position of that sensor, i.e. for localisation. In this thesis we present three studies that take us towards precise localisation. Paper I involves finding exact — on a subsample level — TDoA measurements. These types of subsample refinements give a higher precision, but are sensitive to noise. We present an explicit expression for the variance of the TDoA estimate and study the impact that noise in the signals have. In Paper III TDoA measurements are used to estimate sender and receiver positions in an efficient way. We present a new initialisation approach followed by a scheme for performing local optimisation for TDoA data with constant offset, i.e. when the sound events are repetitive with some constant period. The sender and receiver positions together constitute a map of the environment and such maps are studied in Paper II. Assuming that we have a number of different map representations of the same environment — coming from either sound, radio or image data — we present an algorithm for how to merge these representations into one map, in an efficient way using only a small memory footprint representation. The final map has a higher precision and the method can also be used to detect changes that have occurred between the creation of the different map representations. Thus, altogether, we present a number of improvements of the localisation process. We perform analysis as well as experimental evaluation of each of these improvements.

List of Publications

This thesis is based on the following papers:

Main papers

I **Stochastic Analysis of Time-Difference and Doppler Estimates for Audio Signals**

G. Flood, A. Heyden, K. Åström
International Conference on Pattern Recognition Applications and Methods (ICPRAM). Springer, Cham, 2018.

II **Efficient Merging of Maps and Detection of Changes**

G. Flood, D. Gillsjö, A. Heyden, K. Åström
Scandinavian Conference on Image Analysis (SCIA), 2019.

III **Robust Self-Calibration of Constant Offset Time-difference-of-arrival**

K. Batstone, G. Flood, T. Beleyur, V. Larsson, H. R. Goerlitz, M. Oskarsson, K. Åström
IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.

Subsidiary papers

IV **Estimating Uncertainty in Time-difference and Doppler Estimates**

G. Flood, A. Heyden, K. Åström
7th International Conference on Pattern Recognition Applications and Methods (ICPRAM), 2018.

Author's Contributions

I The paper evolved from a previous paper written by KÅ and AH. I contributed to the theory development and discussions about the paper. I wrote most of the code for the experiments and generated the final results. I also did a large part of the writing, with input from the co-authors.

II The idea for the merging algorithm came from KÅ, but I contributed to the development of the original idea to the theory and methods used in the paper. The majority of the code was written by me and DG. I did most of the writing of the paper, but all the authors contributed, and wrote some parts each.

III The original ideas were suggested by KÅ, but MO, KB and VL help to develop them for the paper. The coding was mainly conducted by KÅ, MO, KB and VL and VL developed the fast solver. I contributed to some of the experiments, both in the collection of real data and parts of the synthetic experiments. I also took part in the writing process together with all the other authors.

Acknowledgements

I would like to express my sincere gratitude to my main supervisor Kalle Åström and my co-supervisors Anders Heyden and Carl Olsson. Thank you for guiding me into this jungle of academia. Thank you for pushing me when I have been in need of that, and encouraging me at all points. Also, I highly appreciate your help with this thesis — from supporting the idea, despite the late notice, to proofreading. I am also grateful to the co-authors of the papers, for collaborations as well as valuable discussions. To my other colleagues at the Centre for Mathematical Sciences, and especially my fellow PhD students: thank you for making hard times easier and good times even better. Finally I would like to thank my family, and especially Johan; thank you for always being by my side, for your endless patience, help and understanding.

Funding

This work was supported by the Swedish Foundation for Strategic Research project — Semantic Mapping and Visual Navigation for Smart Robots — (grant no. RIT15-0038). It was also partially supported by strategic research project ELLIIT and Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

Contents

Chapter 1

Introduction

Recently, audio and radio sensors have become more used in everyday tools, such as PC:s, laptop and smartphones. If the locations of the sensors are known, they can form a sensor network that can be used for localisation in and mapping of the environment [4, 7, 6, 11]. These two things will be the main topics of this thesis. Even if all the sensor positions are unknown, both sender and receiver positions can be determined up to a choice of coordinate system [9, 18, 21, 30, 43]. These calculations require either time of arrival (TOA) or time-difference of arrival (TDoA) measurements, i.e. measurements of the travel time from each sender to each receiver, or the differences in travel time from a sender to two receivers, respectively. The two types of measurements can also be thought of as absolute and relative time distances. Other usages of sensor networks are, among others, sound quality improvement using beam-forming [1], and speaker diarisation [2].

TDoA measurements can for example be obtained by correlating two received signals, where the TDoA estimation is given by the shift that maximises the correlation function. The better these estimates are, the better will the precision of the applications be. Furthermore, if the TDoA values are used for localisation, the result is also highly dependent on the initialisation. One way to find initial estimates for the positions is to solve a large system of equations, where the equations describe the distances between the sensors and the connections between the TDoA measures [22, 37].

When we use these techniques for several TDoA measures we get a map of the environment, but one can also compute the trajectory for a moving sensor in the scene. For vision, the process of estimating map parameters and sensor motion using only images is referred to as structure from motion (SfM) and simultaneous location and mapping (SLAM) [10, 12, 16, 40]. The resulting map consists of a set of points, which each have a position and a feature vector. For sound and radio data a similar approach can be used, and sometimes this has been referred to as structure from sound [38]. If the measurement errors are zero mean and Gaussian, the maximum likelihood estimate of the map is obtained by minimising the sum of squared errors, where the errors are modelled by some error function connected to the type of sensor data. The iterative process that is used for finding these estimates is within photogrammetry and computer vision referred to as bundle adjustments [39].

A successful bundle adjustment and SfM outcome for vision — one that gives a good map of the environment — requires a large amount of images. And actually, it is not only the use of audio and radio sensors that has increased over the last couple of years, but also the availability of cheap and good cameras. Today, there is a decent camera in every mobile phone, every laptop and every new car. This allows for fast data collection using crowdsourcing, which in turn requires faster algorithms to handle all data. One example could be the following: Imagine that every car driving through a city could build its own map of the city using its collected image data. The result would be a large amount of representations of the same city map. If we have a fast and accurate way to merge individual map representations into one global map, each car could contribute to that global map. The city map would thus improve with each car passing.

The problem of map merging has a strong connection to loop closure, where any drift that has arisen in the SfM process has to be adjusted for when the camera returns to a position that has already been visited [42]. This corresponds to a merge of the start and the end of the map. The issue has also been addressed within the field of collaborative SLAM, where several cameras are simultaneously used for SLAM. There are for example applications with several drones where the merge is based on a few keyframes [34]. This is fast, but is highly dependent on the choice of keyframes. In some other cases the problem has been simplified by common initialisation [44], or by mounting the cameras on a platform [28].

In this thesis we investigate several ways of achieving precise localisation. The three included papers are ordered according to their publication dates, but following a logical order we would first look at Paper i, followed by Paper iii and lastly Paper ii. In Paper i we study how to find exact TDoA estimates from received sound data and we also present a stochastic analysis to explain the precision of such estimates. Paper iii uses TDoA measures to both initialise and optimise a map of the sensors involved. Finally, Paper ii explains how a number of map realisations of the same scene can be merged to obtain a global map of lower variance.

Before the papers are presented, the introductory part of the thesis will be organised as follows: In Chapter 2 some methodology will be introduced and in Chapter 3 we go through sensor modelling. In Chapter 4 we show how sensor data can be used for mapping and localisation. Chapter 5 treats the problem of map merging and in Chapter 6 we present some results from the three papers and discuss how this work can be developed in the future.

Chapter 2

Methodology

This thesis is devoted to the study of signals. A signal can be described as the entity that carries some sort of information from one point to another [17, p. 1]. One can also view it as any physical quantity that varies with time, space or one or several other independent variables [31, p. 2]. A large part of this thesis regards the study and analysis of audio, i.e. information transmitted as an acoustic wave through space. This is an example of a one-dimensional signal. Other examples of such are radio signals, e.g. ultra wide-band (UWB), which is used in Paper III. UWB is a radio technology that transmits information short distances using a wide frequency band and low energy [36]. There are also signals of higher dimensions than these, such as two-dimensional images.

Even if we study several different signal realisations, many principles are the same for them all and both methods and applications can be adjusted slightly such that another signal type can be used. Therefore, parts of this thesis will discuss signals in general, while parts will be more focused on for example audio and images.

1 Sampling, Interpolation and Smoothing

Most of the signals that we model and analyse are in reality analog, but for analysis they need to be converted into a digital format. For the one-dimensional case this is done through sampling. Assume that we have an analog signal $x_a(t)$ and denote the discretisation operator by $D : \mathbb{B} \to \ell$. Here, $\mathbb{B}$ are functions $f \in C(\mathbb{R}, \mathbb{R})$ that are square integrable with vanishing Fourier transform outside $[-\pi, \pi]$, while ℓ denotes the set of discrete, square integrable functions from $\mathbb{Z}$ to $\mathbb{R}$. An analog signal can thus be sampled by

$$x(n) = D(x_a)(n) = x_a(nT), \tag{2.1}$$

where T is the period, i.e. a sample is taken every T seconds, and $x(n)$ is the digital signal corresponding to $x_a(t)$. The sampling period also defines the sampling frequency, $F_s = 1/T$.

As long as the sampling frequency is at least twice as big at the highest frequency $F_{\max}$ in the signal, $F_s > 2 \cdot F_{\max}$, the *sampling theorem* states that the analog $x_a(t)$ can be recovered exactly from its digital representation $x(n)$ [20, 29, 31, 35, 41]. The analog signal $\hat{x}_a$ can be obtained by interpolation of x with a kernel g. If we denote the interpolation operator $I_g : \ell \to \mathbb{B}$, we have

$$\hat{x}_a(t) = I_g(x)(t) = \sum_{i=-\infty}^{\infty} g(t-i)x(i). \tag{2.2}$$

With g given by the normalised sinc operator

$$\mathrm{sinc}(k) = \frac{\sin(\pi k)}{\pi k}, \tag{2.3}$$

we have that

$$I_{\mathrm{sinc}}(D(x_a)) = x_a. \tag{2.4}$$

Hence, $\hat{x}_a(t) = x_a(t)$ if $\hat{x}$ is obtained using $g = \mathrm{sinc}$. This is referred to as ideal interpolation.

In many cases a sampled signal does however also contain noise. Then, the connection could rather be described as

$$\tilde{x}(n) = x(n) + e(n) = D(x_a)(n) + e(n), \tag{2.5}$$

where $e(n)$ describes the noise and $\tilde{x}(n)$ is the digital signal that is used for analysis. The noise can have several origins, such as other signals that were not supposed to be captured or added noise that has arisen in the sampling process. For an audio signal this could be someone speaking in the background, or disturbances in the microphone.

Noise coming from disturbances often has a high frequency. To remove some of that noise one can apply a filter to the sampled signal, in order to smooth out the high-frequency components. Usually the signal itself contains lower frequencies, so most of the information in the signal can be kept through the filtering. Furthermore, patterns on a coarser scale are easier captured after smoothing [25]. In this thesis, smoothing will refer to interpolation with the Gaussian kernel

$$G_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{x^2/(2\sigma^2)}. \tag{2.6}$$

The value σ is the standard deviation of the Gaussian and determines the width of the kernel.

Furthermore, there is another advantage of smoothing with a Gaussian when ideal interpolation is employed. It turns out that interpolation with a Gaussian kernel followed by interpolation with the sinc kernel can be approximated by only Gaussian interpolation [3]. What we get is

$$\hat{x}_a(t) = (G_\sigma * I_{\text{sinc}}(x))(t) = I_{G_\sigma * \text{sinc}}(x)(t) \approx I_{G_\sigma}(x)(t). \tag{2.7}$$

However, this only holds when the standard deviation σ is *large enough*. How large it has to be is studied in Paper 1.

2 Bundle Adjustment

The term *bundle adjustment* originates from the field of photogrammetry [15], and has later been widely used within computer vision [16]. The name refers to the bundle of light rays that goes from points in space to each camera. The process consists of simultaneous optimisation of the cameras and the structure of the scene. In the case of computer vision, this would refer to the camera positions, their intrinsic parameters and the 3D feature points. The method can also be translated to one-dimensional signals, and for sound signals the optimisation would be performed over the senders as well as the receiver positions. In [39], the method is described as: "Bundle adjustment is really just a large sparse geometric parameter estimation problem".

Classically, bundle adjustment is formulated as a non-linear least squares problem. Within this thesis, this is also what has been used, but it is worth mentioning that there does exist a number of variations where the cost models are non-quadratic, see [39].

If we collect the parameters which we want to optimise in θ and the values that can be measured in β, we want to minimise the distance from the measured values β_m to the by θ estimated values $\beta_e(\theta)$. This gives the error function

$$f(\beta_m, \theta) = \sum_{i=1}^{n} d(\beta_e(\theta_i), \beta_{mi})^2, \tag{2.8}$$

where n is the number of measured values and d is a distance function, often given by the Euclidean norm $||\cdot||_2$. This problem is large, non-linear and can be solved by some method for non-linear optimisation, such as Gauss-Newton [14, 23], or Levenberg-Marquardt [24, 27]. To obtain the optimal parameters,

$$\theta_{\text{opt}} = \text{argmin}_\theta f(\beta_m, \theta), \tag{2.9}$$

we perform iterative minimisation of the error function in Equation (2.8). When the measurement errors are zero mean Gaussian the simplest way to express the optimal parameter update in each iteration step is by

$$\Delta\theta = -(J^T J)^{-1} J^T \mathbf{r}, \tag{2.10}$$

with $\mathbf{r}$ being a vector of all the residuals, i.e. all the terms in f, and J the corresponding Jacobian. The equation shows the update for the Gauss-Newton method. Levenberg-Marquardt is an extension of this, with the update described by

$$\Delta\theta = -(J^T J + \lambda I)^{-1} J^T \mathbf{r}, \tag{2.11}$$

where I is the identity matrix and λ is a non-negative scalar which follows some updating scheme.

For the bundle adjustment to give a good result, a good initialisation is required. However, this is a research area in itself and will not be covered in the introductory part of the thesis. In two of the papers, we assume that a good initialisation is already found, while we present a solution for the initialisation in Paper III.

3 RANSAC

A big challenge that arises when working with real data is the handling of outliers. In the error model in Equation (2.8) each of the n measurements is equally important and if a few measurements are wrong, that will be highly penalised. In the presence of outlier values, a parameter estimation that in reality is good can give a large error, which would make us discard that estimation. Actually, many big outliers may prevent us from finding any good solution to the optimisation problem at all.

One way to get past this problem is to use the *RANdom SAmpling Consensus* (RANSAC) [13]. The idea with RANSAC is to use as little data as possible to estimate the model parameters and then use the rest of the data to evaluate these parameters. If m is the minimum number of data points that are needed to estimate parameters for the chosen model, the algorithm works as follows:

Algorithm 1: RANSAC

```
while Model is not good enough do
    Randomly select an initial guess of $m$ data points
    Estimate the model parameters from the selected set of points
    Count how many of the other data points that are close enough to the
      estimated model (this is the consensus set)
end
The model is good enough, keep the model and (potentially) improve it using all of
  the consensus set.
```

In the instructions above, we have not defined what *close enough* means for the consensus set and how to know that a model is *good enough* to terminate. These are parameters for the RANSAC method that have to be decided. A maximum number of iterations is also required in order to not get stuck when an optimum cannot be found.

The RANSAC method is specifically good when the set of outliers is large. In this case, just optimising over the whole dataset will not yield a satisfying result, while RANSAC can provide a good solution and detect the outliers as well [13]. Since the first paper about RANSAC was published, a number of variations have been presented and today these are also widely used. For some of them, see [5, 19, 32].

Chapter 3

Sensor Modelling

In this chapter details about sensor modelling are presented and the analysis of signals is developed. First, the focus will be on sound, but the same principles often apply to other signals, such as UWB. Later, we will also present some modelling equations for images. First, we will focus on measurements that can be used to calculate sender and receiver positions and thus can be used for localisation.

1 TOA and TDoA

Assume a setup with a number of senders, $s_i \in \mathcal{R}^3$, $i = 1, .., m$ and receivers $r_j \in \mathcal{R}^3$, $j = 1, .., n$. Also, assume that these are synchronised, i.e. that their internal clocks coincide. Then, by comparing when the signals were emitted from the senders to when they reached the receivers, we can get the travel time, and by multiplying these measurements with the speed of the signal v, the absolute distance measures between each sender and receiver can be derived. The distance will be

$$d_{ij} = v(t_{ij} - T_i) = \|r_j - s_i\|, \tag{3.1}$$

where t_{ij} denotes the arrival time for signal i to receiver j and T_i is the emission time. This time difference is called the time of arrival measurement (TOA) or absolute travel time, see [37].

One could also consider a setup where the senders are synchronised and the receivers are synchronised, but not to each other. That would instead give us time-difference of arrival measurements (TDoA), or relative travel time; for emitted sound, we know how much longer it took for the sound to reach receiver 2 compared to receiver 1, etc. This could be

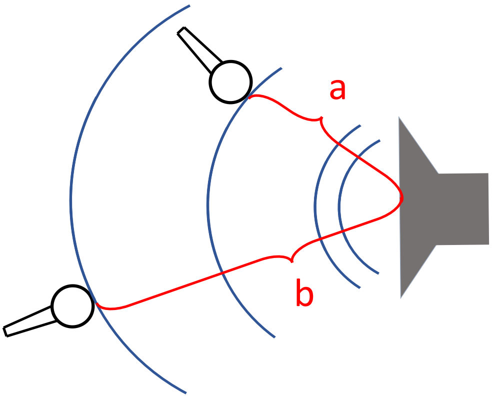

Figure 3.1: The figure is illustrating TOA and TDoA measures. The sound emitted from the speaker will reach the two microphones at different times. For TOA we have that $d_{11} = a$ and $d_{12} = b$. For TDoA we instead measure the difference, $(t_{12} - T_1) - (t_{11} - T_1) = (b - a)/v$.

explained by the following equation

$$z_{ij} = \|r_j - s_i\| + o_i. \tag{3.2}$$

Here, o_i is an offset that is different for each sound event and the arrival and emission times are not synchronised. What we actually measure in this case is

$$(t_{ij} - T_i) - (t_{ik} - T_i) = t_{ij} - t_{ik}, \tag{3.3}$$

i.e. how much longer it took for sound i to reach receiver j than receiver k. The value can be multiplied by v to give the difference in distance. Equation (3.2) can be obtained from

$$\|r_j - s_i\| = v(t_{ij} - T_i) = vt_{ij} - vT_i. \tag{3.4}$$

Now, we add and subtract the constant $T_0 = t_{ik}$ from this equation, which gives

$$\|r_j - s_i\| = v(t_{ij} - T_0 + T_0 - T_i) = v(t_{ij} - t_{ik}) + v(T_0 - T_i). \tag{3.5}$$

Note that the receiver index k in t_{ik} is fixed. Since v is known and $t_{ij} - t_{ik}$ can be measured, we call that term $z_{ij} = v(t_{ij} - t_{ik})$, while the part that cannot be measured will be the offset $o_i = -v(T_0 - T_i)$, which results in Equation (3.2). For an illustration of TOA and TDoA, see Figure 3.1.

There are also other variations of TOA and TDoA. One can think of a situation where the senders are not synchronised with the receivers, but where the signals are emitted regularly. This would give an expression similar to Equation (3.2), but with a constant offset, i.e.

$$z_{ij} = \|r_j - s_i\| + o. \tag{3.6}$$

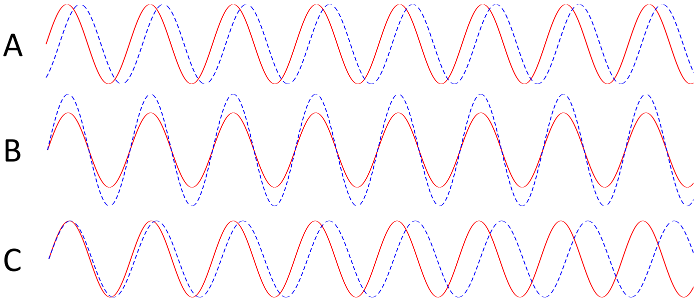

Figure 3.2: This shows what different sound models can capture. In A we have a sound and a translated copy of it, connected to Equation (3.7). Plot B shows what happens if we only have the impact of an amplitude, corresponding to γ, with $h = 0$, in Equation (3.8) and C shows the same thing but for Doppler, corresponding to $h = 0$, $\gamma = 0$ in Equation (3.9).

Paper I in this thesis uses regular TDoA measurements, while Paper III focuses on the constant offset problem. Note that by subtracting the offsets o_j or o from z_{ij} we get the distance measurements d_{ij}.

2 Model Selection and Parameter Estimation

To find the optimal model parameters using Equation (2.9) the model must first be decided. Assume that we have two measurements from different receivers, where the received signals $x(n)$ and $\bar{x}(n)$ come from the same emitted signal. The easiest connection between these two is to consider one of them to be a translated version of the other,

$$x(n) = \bar{x}(n + h). \tag{3.7}$$

This corresponds to TDoA, where h would be the time difference value in Equation (3.3). However, the connection between the signals could also look differently. For example, it is reasonable to assume that the signal is stronger when it reaches a receiver that is close to the sender, compared to one that is further away. Therefore, an amplitude parameter γ might be added

$$x(n) = \gamma\bar{x}(n + h). \tag{3.8}$$

Furthermore, if the sender is actually moving while emitting the sound this may result in a stretched or compressed signal. This raises the need of a Doppler parameter α,

$$x(n) = \gamma\bar{x}(\alpha n + h). \tag{3.9}$$

The effect of a translation, an amplitude difference and the presence of a Doppler factor is shown in Figure 3.2. The models in Equations (3.7), (3.8) and (3.9) have been used in Paper

1. One could also think of a number of other suitable models, and the model might need to be adjusted to the specific problem.

The above models could be used to express an optimisation problem, similar to the one in Equation (2.9), if we collect the parameters of interest in θ. For the model in (3.7) we would have $\theta = \{h\}$ and for the model in (3.9) we have $\theta = \{h, \gamma, \alpha\}$, while $\beta = \{x, \bar{x}\}$ in both cases.

3 Estimating TOA and TDoA

Once the model has been decided, the error function from Equation (2.8) can be used to estimate the parameters θ. Again, assume that we have two signals x and $\bar{x}$ and that we want to find how these relate to one another. In this case, we can express the error function as

$$f(\beta, \theta) = \sum_n (x(n) - \bar{x}(\eta(\theta)))^2, \tag{3.10}$$

where $\beta = \{x, \bar{x}\}$ and $\eta(\theta)$ can be either n, $n+h$ or $\alpha n+h$, in accordance with the previous section. Comparing this to the general error function (2.8) we see that the distance function d is given by the difference between the two signals. Once this error function is formulated, the parameter estimation can be found using the methods previously described.

If x is the emitted signal, $\bar{x}$ is the received signal and the model is that of Equation (3.7) the estimated value h will represent the time of arrival. If we instead choose x and $\bar{x}$ to be the signals received by two different receivers coming from the same emitted signal, we will estimate a time-difference of arrival.

Furthermore, if we use the model in Equation (3.7) we can also estimate the parameter h using cross-correlation. The cross-correlation for real signals x and $\bar{x}$ is defined as

$$(x \star \bar{x})(h) = \sum_n x(n)\bar{x}(n+h). \tag{3.11}$$

The translation h is obtained by maximising the cross-correlation function,

$$h_{opt} = \text{argmax}_h (x \star \bar{x})(h). \tag{3.12}$$

This estimation will be exactly the same as the one coming from minimisation of (3.10), since

$$
\begin{aligned}
\operatorname{argmin}_h f(\beta,\theta) &= \operatorname{argmin}_h \sum_n (x(n) - \bar{x}(n+h))^2 = \operatorname{argmin}_h \sum_n (x(n))^2 + \\
&(\bar{x}(n+h))^2 - 2x(n)\bar{x}(n+h) = \operatorname{argmin}_h \sum_n -2x(n)\bar{x}(n+h) \\
&= \operatorname{argmax}_h \sum_n x(n)\bar{x}(n+h) = \operatorname{argmax}_h (x \star \bar{x})(h).
\end{aligned} \tag{3.13}
$$

Hence, for the models in Equations (3.8) and (3.9) we use error function (3.10), but if we stay with the smaller model (3.7) we can use cross-correlation to find the h that minimises the error function. There are also variants of cross-correlation, e.g. GCC-PHAT, which we used for initialisation in Paper I. For more information, see [18].

3.1 Estimation on Subsample Level

In the equations above, the translation h will be estimated as an integer number of samples. To refine the estimations further, we can do the parameter estimation on continuous signals, achieved from ideal interpolation. This would result in the following error function

$$
f(\beta,\theta) = \int_t (x_a(t) - \bar{x}_a(\tau(\theta)))^2 \, \mathrm{d}t, \tag{3.14}
$$

with τ defined in accordance with η above, but for continuous values. Minimising this error would result in parameters on an even finer scale than the sample rate, and thus even more exact measures. However, the estimation also becomes more sensitive to noise. This matter is considered in Paper I.

4 Sensor Modelling for Vision

For computer vision and 3D reconstructions the pinhole camera model is a common way to model the relation between the 3D points, 2D points and the camera matrices [16]. Two alternatives are the affine and projective camera models.

4.1 Camera Models

The pinhole camera model takes a point $U = \begin{bmatrix} X & Y & Z \end{bmatrix}^T$ in 3D to a point $u = \begin{bmatrix} x & y \end{bmatrix}^T$ in the image by following the straight line from U to the camera center in the origin, $c = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, and intersecting it with the image plane. If the distance from the camera

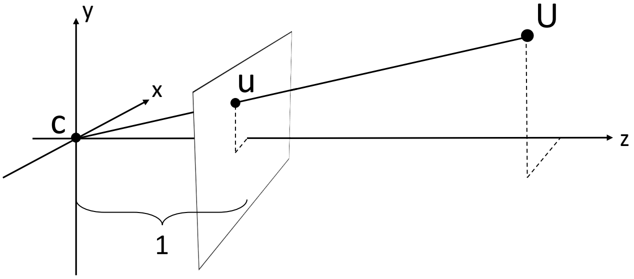

Figure 3.3: An illustration of the pinhole camera model. The camera centre c is located in the origin in the xyz-coordinate system. The image plane is parallel to the xy-plane and located at unit distance from c in the positive z-direction. A 3D point U is mapped to the image point u, where u is given by the intersection of the image plane and the straight line going from U to c, see Equation (3.15).

centre to the image plane is 1 and we assume that the image plane is parallel to the xy-plane, the projection is given by

$$u = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} X/Z \\ Y/Z \end{bmatrix}, \tag{3.15}$$

see Figure 3.3.

For camera modelling it is convenient to describe the points in 2D and 3D using homogeneous coordinates, i.e. we write u as $\hat{u} = \begin{bmatrix} x & y & 1 \end{bmatrix}^T$ and represent U by $\hat{U} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$. For a general camera projection matrix $P \in \mathbb{R}^{3\times 4}$, we have the relationship

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{3.16}$$

where we then can divide by c to obtain the image point in homogeneous coordinates, $\begin{bmatrix} a/c & b/c & 1 \end{bmatrix}^T = \begin{bmatrix} x & y & 1 \end{bmatrix}^T$. We usually write this as a proportionality,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{3.17}$$

For the pinhole camera model the camera matrix is $P = \begin{bmatrix} I & 0 \end{bmatrix}^T$, with I being the 3×3 identity matrix, and 0 a 3×1 vector of zeros. If the camera moves, that can be represented by a 3×3 rotation matrix R (i.e. $R^T R = I$ and $\det(R) = 1$) and a 3×1 translation vector t, giving $P = \begin{bmatrix} R & t \end{bmatrix}^T$. These are sometimes referred to as *extrinsic parameters*.

A more general model of the camera matrix is

$$P = K\begin{bmatrix} R & t \end{bmatrix} = \begin{bmatrix} \mu f & s & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & t \end{bmatrix}. \tag{3.18}$$

The matrix K describes the *intrinsic parameters* of the camera. The value f is called the focal length and is a re-scaling parameter. Both the aspect ratio μ and the skew parameter s re-scales for non-square pixels and the point $\begin{bmatrix} x_0 & y_0 \end{bmatrix}^T$ is referred to as the principal point, and gives a translation. A camera represented by the matrix in Equation (3.18) is called a finite projection camera.

We call P and affine camera if it has the structure

$$P = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix}, \tag{3.19}$$

where A is a 2×3 matrix, t has size 3×1 and 0 is 1×3 vector of zeros. The affine camera is a good approximation when the distance from the camera to the scene is much larger than the depth of the scene. A general projection camera is given by an arbitrary 3×4 matrix with rank 3 [16].

4.2 Parameter Estimation for Vision

When it comes to vision, we can always measure the images, and thus the image points, while the 3D points and the camera matrices might be known or unknown. Assume that we have a set of m cameras which each capture n different 3D points. If both 3D points and cameras are unknown, our sought set of parameters will be $\theta = \{P_1, ..., P_m, U_1, ..., U_n\}$. The measurable quantities will be the image points that comes from projecting each 3D point in each camera, $\beta = \{u_{11}, ..., u_{1n}, u_{21}, ..., u_{mn}\}$, where $\hat{u}_{ij} \sim P_i\hat{U}_j$ and β in total contains $2mn$ values, since each u_{ij} has 2 unknown coordinates. Remember that $\hat{u}_{ij}$ and $\hat{U}_j$ are the honomgeneous representations of u_{ij} and U_j, respectively. If the 3D points or the cameras are known, we move these parameters from θ to β. In either case, the error function can be described as

$$f(\beta, \theta) = \sum_{ij} \|\hat{u}_{ij} - \frac{P_i\hat{U}_j}{\lambda_{ij}}\|^2. \tag{3.20}$$

The values λ_{ij} are proportionality constants that arise when we make $P_i\hat{U}_j$ homogeneous.

Chapter 4

Mapping and Localisation

1 Multilateration for TOA

Multilateration describes the process of determining the position of a sender given the distances to several receivers. These distances can for example come from TOA or TDoA measurements. To illustrate the idea, we consider two receivers r_1, r_2 and one sender s_1, all located in a plane. If the distances d_{11} and d_{12} from the sender to each of the receivers are known, there are two potential points where the sender could be located, namely where the circle with centre in r_1 and radius d_{11} and the circle with centre in r_2 and radius d_{12}

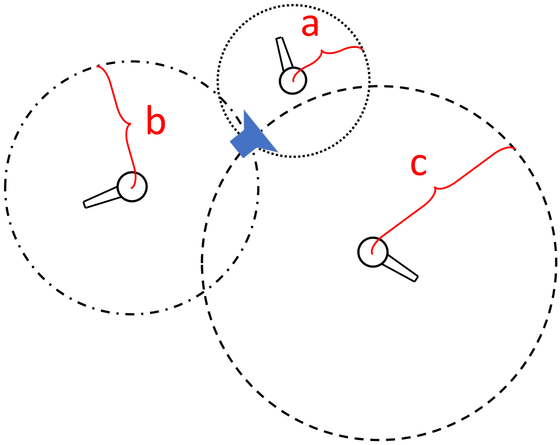

Figure 4.1: The figure shows multilateration in a plane using one sound source and three receivers. The location of the speaker is unknown and the positions of the microphones, r_1, r_2, r_3 are known. Furthermore, the distances $d_{11} = a$, $d_{12} = b$ and $d_{13} = c$ are measured. The location of the loudspeaker is given by the intersection of the three circles.

intersect. If we add a third receiver r_3, the circle around that with radius d_{13} will contribute to a single intersection point of the three circles and thus the solution for the position of s_1. For an illustration, see Figure 4.1.

In three dimensions each distance measure d_{ij} defines the radius of a sphere with r_j as centre, and in a similar way the positions of the senders can be found as the intersections of the different spheres. If we have few measurements, we might get several possible solutions. The opposite problem, when senders are known and receivers and unknown, is solved identically, i.e. it does not matter for the algorithm whether a *node* is a sender or a receiver. In the presence of noise, the system has to be be solved in a least squares sense.

1.1 System of Equations

From the TOA or TDoA measurements we can formulate a large system of equations, representing the spheres we discussed in the previous section. We could have a static setup where the receiver positions are unknown, but one wants to locate the sender positions, as in the small 2D example above. Actually, given enough measurements, it is possible to calculate both receiver and sender position, if all are unknown [22, 38]. Independently of which case we are looking at, the following will be true.

If we let i denote the sender number and j the number of the receiver, the TOA case will yield the system of equations

$$d_{ij}^2 = \|r_j - s_i\|^2, \tag{4.1}$$

and in the case of TDoA we get

$$(z_{ij} - o_i)^2 = \|r_j - s_i\|^2, \tag{4.2}$$

for all i and j. If we have m senders and n receivers this will result in mn equations. An initial guess for the unknowns can be found with methods from algebraic geometry using Groebner bases, see [8, 37].

Once the initial guess is found, the error function (2.8) can be formulated as

$$f(\{d_{ij}\}, \{r_j, s_i\}) = \sum_{i,j} (d_{ij}^2 - \|r_j - s_i\|^2)^2, \tag{4.3}$$

for TOA and

$$f(\{z_{ij}\}, \{r_j, s_i, o_i\}) = \sum_{i,j} ((z_{ij} - o_i)^2 - \|r_j - s_i\|^2)^2, \tag{4.4}$$

for TDoA. After this, bundle adjustment can be performed to minimize the error and to achieve optimal estimates.

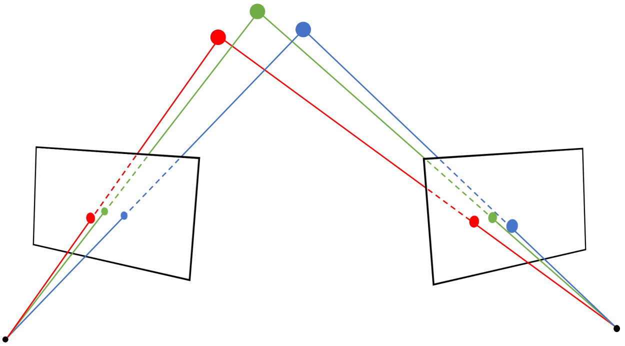

Figure 4.2: Triangulation of three points given two images. The point pairs u_i, $\bar{u}_i$ as well as the cameras P and $\bar{P}$ are given. Each 3D point U_i is obtained by intersecting the lines from each camera centre through the corresponding image point.

2 Triangulation for Image Data

Triangulation refers — just as the name suggests — to deciding distances using triangles. In computer vision, this can be used to find 3D points and it is the correspondence to multilateration for sound.

Assuming that we have two images u and $\bar{u}$ taken by the cameras P and $\bar{P}$, respectively, we for each corresponding feature point pair j have,

$$u_j = PU_j \quad \text{and} \quad \bar{u}_j = \bar{P}U_j, \tag{4.5}$$

where U_j is the unknown 3D feature point that the image points u_j and $\bar{u}_j$ depict. If the camera matrices P and $\bar{P}$ are known, one can draw a line from each camera centre, through the corresponding image point, out in space. The two lines will lie in a plane and intersect in a point — the point U_j [16]. If this is done for many points and images you finally get a map of the environment. A small triangulation example is shown in Figure 4.2. To find corresponding image points $\{u_j, \bar{u}_j\}$, one can for example use SIFT [26], or ORB features [33].

In reality, there is in most cases some noise in the measurements and the 3D points will have to be estimated in a least squares sense instead. Furthermore, the camera matrices are often unknown as well. Both estimates of the cameras and 3D points can be obtained using bundle adjustment.

2.1 Localisation in a Known Environment

For TOA and TDoA problems, the algorithms for mapping the environment and localising a new sensor in the scene are identical. The reason for this is that senders and receivers are represented in the same way. However, this is not the case for vision. A scene can be mapped using the triangulation described in the previous section on several images. In that case, the image points and the cameras are known, while 3D points are unknown. When it comes to localisation, we have the opposite problem: the image points and the 3D points are known, while the cameras are unknown. The problem is solved by finding a solution to the system of equations given by

$$\lambda_j u_j = P U_j, \qquad j = 1, ..., n, \tag{4.6}$$

where the camera matrix P is the sought value and λ_j are unknown factors that arise when we make the image points homogeneous. This corresponds to the error function in Equation (3.20), but with only one camera matrix. As long as we have sufficiently many point correspondences u_j and U_j — at least six — we can solve for the unknowns using a method called direct linear transform (DLT). If we are interested in the intrinsic parameters of the camera, we first find P and then factorise it to obtain K, see [16] for details.

3 Structure from Motion

We have previously in this chapter explained how mapping and localisation can be done using sensor data. For sound and radio data localisation corresponds to finding a sender position s_i and mapping is done by finding the receivers r_j. The correspondence in vision is finding the cameras P_i for localisation and the 3D points U_j for mapping. When both mapping and localisation are performed at once, this is within the field of computer vision referred to as structure from motion (SfM) or simultaneous location and mapping (SLAM). This section, however, treats both vision, sound and radio, since the principles are the same for other type of sensor data than images.

For audio and radio the TOA or TDoA measures are the known sensor data, while both sender and receiver positions are solved for. In the case of vision, the images and image points are known, while both 3D feature points and camera matrices are unknown. Such systems can be solved iteratively, using bundle adjustment. The error functions will be expressed according to Equations (3.10) and (3.20). The difference to previous cases will be which parameters that are contained in the set of unknowns θ and and which in the set of known parameters β. This will in turn change the Jacobian used to find the parameter update in Equations (2.10) and (2.11). In many cases when we want to use vision to map an environment we need to use SfM algorithms, since it is uncommon that the camera positions are known. However, once the map is obtained we can use it for localisation.

Chapter 5

Map Merging

The previous chapters have described mapping and localisation and some methods needed for that. In this chapter we will discuss one matter that can improve the localisation in a known environment. Except for a good algorithm and exact measurements, the localisation is highly dependent on the description of the environment, i.e. the map. A map consists of a number of feature points and each feature point has a descriptor and a position. In the case of TOA or TDoA each point would be a receiver, with an ID as the descriptor and the position in 3D as the position. For image data, the position would be a 3D point as well, while the descriptor could be a SIFT descriptor [26], ORB feature [33] or correspondingly. The map would be the set of all such feature points.

The more exact our map is, the better will the localisation be. Furthermore, the more meas-

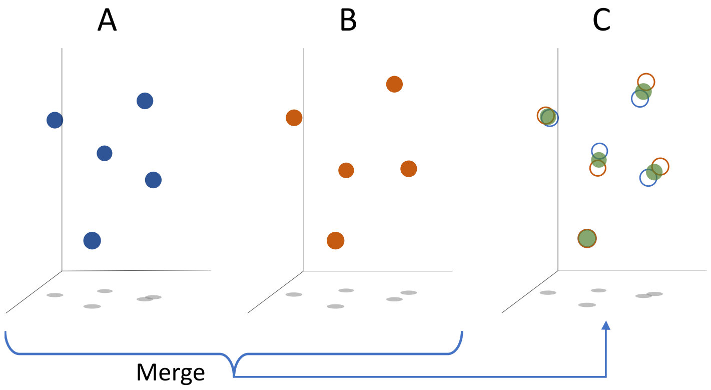

Figure 5.1: A and B shows two different map representations — in blue and red, respectively — of the same environment. However, the corresponding feature points are not exactly the same. In C these are merged into one global map, in green. The positions of the original feature points are shown by contours.

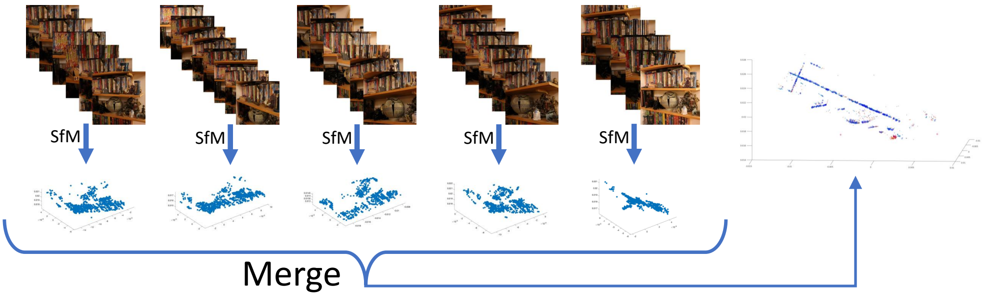

Figure 5.2: An overview of the idea of map merging and how it can be used to improve map quality. First, individual map estimates are created from several images using SfM. Then, these can be merged in order to obtain a global map with lower variance.

urements we have when creating our map, the better will it be. Therefore, the knowledge of how to merge maps is important. If we can merge maps, we can use this to add information — represented as a new map — when we get hold of new information about the environment. Imagine a scenario with self-driving cars in a city. A car is passing though this city; it has a camera and collects a lot of data — enough to build its own local map of the parts that it passes. This data can then be used to contribute to the global map of that city, reducing the variance in those parts of the global map. The next time a car drives through that city, the map would be a bit better.

One straightforward way to add several maps would be to make a new estimation of the environment, using all the data that was used to obtain each of the individual maps. In the case of computer vision, this would correspond to bundling over all 3D points and all camera matrices for each of the local maps at once. However, this process could be time consuming and computationally heavy, since the number of images involved grows quickly. In Paper II, we investigate how map merging can be made more efficient by linearising the residuals. Furthermore, we look into at what cost this would be, i.e. how much information that might be lost in the linearisation.

Map merging has previously been used in e.g. the fields of collaborative SLAM [44] and loop closure [42]. In collaborative SLAM, several cameras are used simultaneously and the problem can be simplified by initialising the cameras such that their relative positions are known. This makes the problem similar to that of loop closure, where the beginning and end of an SfM maps should be merged. In our paper, we have no common initialisation and the mappings are assumed to be done at different times. An image illustrating the idea of map merging can be seen in Figure 5.2 and a small example of map merging is shown in Figure 5.1.

Chapter 6

Conclusions

The common theme for the papers included in this thesis is the goal to achieve precise localisation. A majority of the work has been focused on sound and UWB signals, but can in many cases also be applied to images or other types of signals.

In Paper I, we concentrate on the details. Given a good initialisation, the subsample methods that are presented result in estimates with a high precision. We investigate the limits in precision and examine whether a more complicated model that includes more parameters than translation can give an even better result. It turns out that so is the case, but that a larger model than necessary gives a less exact result, i.e. a higher variance. We believe that these explicit formulas for the covariance matrix brings certainty and thoroughness.

Paper III also treats TOA and TDoA problems, and more specifically the TDoA case with a constant offset. We present a novel method for how to efficiently initialise and solve this problem, together with a fast solver for the system of equations in the case of five senders and five receivers. This paper and Paper II concentrate on the speed. The faster the algorithm is, the more iterations can be run and the more data can be used.

As the others, Paper II is primarily devoted to one-dimensional signals. However, it also contains a part where the algorithm is tested on image data. Except for the contribution of a fast algorithm for merging of maps one can also talk about preciseness in the terms of an exact map, i.e. the more exact the map is, the more precise will the localisation be. Since the map merging algorithm is fast, this will in turn bring that more measurements can be used and this gives a better localisation. Furthermore, we show how our method can detect differences between the map realisations. With this we can avoid merging the points that do not match — a situation that can arise when an object in the scene has moved.

Hence, this thesis is an attempt to move towards precise localisation — precise both in terms of exactness and of speed. However, there is still work to be done. The map merging algorithm presented in Paper II needs further focus concerning robustness to outliers and the coordinate system. Another subject for future studies is the initialisation process, since many of the presented algorithms rely on a good initialisation.

Henceforth, we would like to improve the merging algorithm even more, to include a simultaneous estimation of the coordinate system. Another investigation for the future would be to combine these geometric models with modern machine learning techniques to achieve a framework for semantic structure from motion.

References

[1] X. Anguera, C. Wooters, and J. Hernando. Acoustic beamforming for speaker diarization of meetings. *IEEE Transactions on Audio, Speech, and Language Processing*, 15 (7):2011–2022, 2007.

[2] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals. Speaker diarization: A review of recent research. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(2):356–370, 2012.

[3] K. Åström and A. Heyden. Stochastic analysis of image acquisition, interpolation and scale-space smoothing. *Advances in Applied Probability*, 31(4):855–894, 1999. ISSN 0001-8678.

[4] M. Brandstein, J. Adcock, and H. Silverman. A closed-form location estimator for use with room environment microphone arrays. *Speech and Audio Processing, IEEE Transactions on*, 5(1):45 –50, Jan. 1997. ISSN 1063-6676. doi: 10.1109/89.554268.

[5] O. Chum, J. Matas, and J. Kittler. Locally optimized ransac. In *Joint Pattern Recognition Symposium*, pages 236–243. Springer, 2003.

[6] A. Cirillo, R. Parisi, and A. Uncini. Sound mapping in reverberant rooms by a robust direct method. In *Acoustics, Speech and Signal Processing, IEEE International Conference on*, pages 285 –288, April 2008. doi: 10.1109/ICASSP.2008.4517602.

[7] M. Cobos, A. Marti, and J. Lopez. A modified SRP-PHAT functional for robust real-time sound source localization with scalable spatial sampling. *Signal Processing Letters, IEEE*, 18(1):71 –74, Jan. 2011. ISSN 1070-9908. doi: 10.1109/LSP.2010.2091502.

[8] D. A. Cox, J. Little, and D. O'shea. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006.

[9] M. Crocco, A. Del Bue, M. Bustreo, and V. Murino. A closed form solution to the microphone position self-calibration problem. In *ICASSP*, March 2012.

[10] M. Csorba. *Simultaneous localisation and map building*. PhD thesis, University of Oxford Oxford, 1997.

[11] H. Do, H. Silverman, and Y. Yu. A real-time SRP-PHAT source location implementation using stochastic region contraction(SRC) on a large-aperture microphone array. In *ICASSP 2007*, volume 1, pages 121–124, April 2007. doi: 10.1109/ICASSP.2007.366631.

[12] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[13] 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*, 24(6):381–395, 1981.

[14] C. F. Gauss. *Theoria Motus Corporum Coelestium in sectionibus conicis solem ambientium*. Göttingen, 1809.

[15] S. K. Ghosh. *Analytical photogrammetry*. 1988. Second Edition.

[16] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. Second Edition.

[17] M. H. Hayes. *Statistical digital signal processing and modeling*. John Wiley & Sons, 1996.

[18] C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24(4):320–327, 1976. doi: 10.1109/TASSP.1976.1162830.

[19] S. Korman and R. Litman. Latent ransac. In *Computer Vision and Pattern Recognition (CVPR)*, pages 6693–6702, 2018.

[20] V. A. Kotelnikov. On the transmission capacity of ether and wire in electrocommunications. *Material for the First All-Union Conference on Questions of Communication, Izd. Red. Upr. Svyazi RKKA (in Russian). (English translation, PDF)*, 1933.

[21] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström. A complete characterization and solution to the microphone position self-calibration problem. In *ICASSP*, 2013.

[22] V. Larsson, K. Astrom, and M. Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 820–829, 2017.

[23] A. M. Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805.

[24] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

[25] T. Lindeberg. Scale-space theory: A basic tool for analyzing structures at different scales. *Journal of applied statistics*, 21(1-2):225–270, 1994.

[26] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004. ISSN 0920-5691.

[27] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[28] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. Ieee, 2004.

[29] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

[30] M. Pollefeys and D. Nister. Direct computation of sound and microphone locations from time-difference-of-arrival data. In *Proc. of ICASSP*, 2008.

[31] J. G. Proakis. *Digital signal processing: principles algorithms and applications*. Pearson Education, 2007.

[32] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. Usac: a universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(8):2022–2038, 2013.

[33] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, volume 11, page 2. Citeseer, 2011.

[34] P. Schmuck and M. Chli. Multi-uav collaborative monocular slam. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3863–3870. IEEE, 2017.

[35] C. E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37 (1):10–21, 1949.

[36] K. Siwiak. Ultra-wideband radio. *Encyclopedia of RF and Microwave Engineering*, 2005.

[37] H. Stewénius. *Gröbner basis methods for minimal problems in computer vision*. Citeseer, 2005.

[38] S. Thrun. Affine structure from sound. In *Advances in Neural Information Processing Systems*, pages 1353–1360, 2006.

[39] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999.

[40] S. Ullman. The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 203(1153):405–426, 1979.

[41] E. T. Whitaker. On the functions which are represented by the expansions of the interpolation theory, ("theorie der kardinalfunktionen"). In *Proc. Royal Soc. Edinburgh*, volume 35, pages 181–194, 1915.

[42] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardós. A comparison of loop closing techniques in monocular slam. *Robotics and Autonomous Systems*, 57 (12):1188–1197, 2009.

[43] S. Zhayida, F. Andersson, Y. Kuang, and K. Åström. An automatic system for microphone self-localization using ambient sound. In *22st European Signal Processing Conference*, 2014.

[44] D. Zou and P. Tan. Coslam: Collaborative visual slam in dynamic environments. *IEEE transactions on pattern analysis and machine intelligence*, 35(2):354–366, 2012.