Position Detection Based on Intensities of Reflected Infrared Light

Ph.D. Thesis

by

Henrik Vie Christensen

Department of Control Engineering
Aalborg University
Fredrik Bajers Vej 7C, DK-9220 Aalborg Ø, Denmark.

ISBN 87-90664-30-2
November 2005

This thesis was typeset using LaTeX2$_\varepsilon$ in `report` document class.
MATLAB™ is a registered trademark of The MathWorks, Inc.

Preface

This thesis is submitted as partial fulfilment of the requirements for the Ph.D. degree at the Department of Control Engineering, Institute of Electronic Systems, Aalborg University, Denmark. The work has been carried out in the period from August 2002 to November 2005 under supervision by Professor Jakob Stoustrup and Associate Professor Anders la Cour-Harbo.

The Ph.D. project forms the part of the WAVES project which deals with 3D position detection using optical sensors. The purpuose of this work is to use an array of infrared emitters and receivers to estimate the position of a passive object. As part of the Ph.d. work the author has been guest researcher at the Department of Electronics at Alcalá University in Spain. The Waves project is supported by the Danish Technical Science Foundation (STVF) grant no. 56-00-0143.

Aalborg University, November 2005
Henrik Vie Christensen

Abstract

This thesis focus on development of methods for using infrared sensors for position detection. There exists various methods where optical sensors are used to give position estimates, many utilizing multiple cameras. The overall approach in this thesis is to base a position detection sensor system on inputs from an array of infrared emitters and receivers. The signals from the emitters are reflected by an object and captured by the receivers. The attenuation from each emitter via the object to the receivers obviously depends on position, geometrical, and optical properties of the object. The main challenge of this work is to reconstruct the object position based on knowledge of the emitted and received signals.

Methods for reconstructing the object position has been developed for both 3D space and 2D space. The method for position reconstruction in 3D space is based on a 3D Reflection Map Model. The 3D Reflection Map Model is developed to model the intensities of light reflected by a spherical object. The 3D object position is reconstructed by searching the parameter space of the 3D Reflection Map Model. The position reconstruction for the 2D space is based on simple calculations on the direct measurements of reflected light intensities, and includes easy calibration. The method for reconstructing 3D positions has been implemented in a prototype of a “non-Touch Screen” for a computer, so that the user can control a cursor in three dimensions by moving his/hers hand in front of the computer screen. The 2D position reconstruction method is implemented in a prototype of a human-machine interface (HMI) for an electrically powered wheelchair, such that the wheelchair user can control the movement of the wheelchair by head movements. Both “non-Touch Screen” prototype and wheelchair HMI has been tested with success. Furthermore some investigation in using similar array of infrared emitters and receivers for a navigation sensor for a mobile robot has been made. Promising results with this approach has been obtained for modeled light reflections.

Resume

Denne afhandling omhandler udvikling af metoder for brug af infrarøde sensore til positionsbestemmelse. Der eksisterer flere metoder, hvor optiske sensore anvendes til at give positionsestimater, mange af disse benytter kameraer (mindst to). Den overordnede fremgangsmåde i denne afhandling er at basere et positionsbestemmelses sensor system på input fra en række infrarøde sendere og modtagere. Signalerne fra senderen reflekteres af objektet og opfanges af modtagerne. Dæmpningen fra hver sender via objektet til modtagerne afhænger oplagt af position, geometri og optiske egenskaber for objektet. Hovedudfordringen ligger i at rekonstruere objektets position ud fra kendskab til de udsendte og modtagne signaler.

Metoder til at rekonstruere objektets position er blevet udviklet både for 3-dimensionale og 2-dimensionale rum. Metoden til positions rekonstruktion af 3D rum er baseret på en 3D Reflektionskort Model. 3D Reflektionskort Modellen er udviklet til at modellere intensiteten af lys reflekteret af et kugleformet objekt. Den 3-dimensionale position af objektet rekonstrueres ved søgning i parameter mængden for 3D Reflektionskort modellen. Positions rekonstruktionen for 2-dimensionale rum er baseret på en simpel udregning direkte på de målte reflektionsintensiteter, og inkluderer en simpel kalibrering. Metoden til rekonstruktion af 3D positioner er implementeret i en prototype af en "non-Touch Skærm" til en computer, så brugeren kan styre en markør i tre dimensioner ved at bevæge hans/hendes hånd foran computerskærmen. 2D positions rekonstruktionsmetoden er implementeret i en prototype af et menneske-maskine interface (HMI) til en elektrisk kørestol, så kørestolsbrugeren kan styre kørestolens bevægelser med hovedbevægelser. Både "non-Touch Skærm" prototypen og kørestols HMI'en er tested med succes. Yderligere er der undersøgt muligheder for at anvende de samme infrarøde sendere og modtagere til en navigationssensor til en mobil robot. Lovende resultater er opnået for denne anvendelse med modelerede lys reflektioner.

Contents

Chapter 1

Introduction

Making a computer determine the three dimensional position of a physical object in real time, is useful in many applications. In the past different approaches has been made to make a computer do this 3D-position determination e.g. radar systems (long distances) and video cameras and ultrasound transducers for shorter distances.

For short distance video cameras and ultrasound transducers have been used as ’eyes’ for a computer. Video cameras have two major disadvantages when used to determine the position of a physical object in 3 dimensional space. First, two video cameras are needed to make stereoscopic vision so 3D-position determination is possible. Next, the amount of information obtained from the video cameras is huge compared to the few bits of information in the 3D-position e.g. 1000 or more. For this reason a real time 3D-position determination requires a powerful processor to process this information. Considering the price of two video cameras and a powerful processor this is a rather expensive solution. The ultrasound solution does not require the same amount of computational power, but the ultrasound transducers are still quite expensive.

In this thesis the focus is on low-cost infrared sensors for short distance 3D-position determination system that is able to estimate the 3D-position of a passive object. Various types of commercially available infrared sensors exists, ranging from basic light emitting diodes (LED’s) and photodiodes over PIR sensors (motion sensors) to distance measuring sensors, and many others. The basic light emitting diodes and photodiodes are available at very low price, and only need simple additional electronic circuitry to operate. Furthermore the ratio of acquired to desired information for a system with e.g. eight emitters and eight photodiodes for estimation of object position in 3D-space is less than 100. Considering the price of the sensors and the lower amount of acquired data a low-cost 3D-position detection system is expected to be realizable based on infrared LED’s and photodiodes.

Possible applications are a 3D mouse, a ’touch free’ touch screen, and other human-machine interfaces.

1.1 Motivation

The project Wavelets in Audio/Visual Electronic Systems (WAVES) was initiated in 2001. WAVES is a framework programme project from the Danish Technical Research Council, carried out as a cooperation between Aalborg University, The Technical University of Denmark, Bang & Olufsen A/S, LEGO System A/S, Carlo Gavazzi Industri A/S.

WAVES is an extension and an expansion of a previous pilot project ’OPTOCTRL’, co-sponsored between the Danish Technical Research Council, Bang & Olufsen A/S, and Aalborg University. OPTOCTRL dealt with intelligent regulation of optical sensor systems.

The research under WAVES is comprised in five workpackages, one of these concerns 3D position sensors. The Ph.D. project of that workpackage (3D Position Sensors based on Wavelets) is the basis of this thesis.

1.2 Previous and Related Work

In his Ph.D. thesis, on low-cost active sensors [16] Anders la Cour-Harbo proposed to use an array of infrared sensors for detecting the position of a passive object in three-dimensional space as an application for an array of simple, low-cost infrared sensors. A method of mapping reflections to 3D positions are proposed, and a model of the reflection are also proposed (model only for two dimensional space). Further technical and theoretical challenges were stated for future work in order to realize the functionality.

Novotny and Ferrier [25] has used infrared sensors to measure distances in a robot navigation application. Similar approach has been used to in the work of Aytac and Barahan [1, 2], where they use infrared sensors to differentiate and localize target primitives, also for use in robot navigation.

Other attempts use cameras and image processing, e.g. Rheg and Kanade [27], Jennings [12] and Sato et al. [32]. All using multiple cameras in the process of determining and tracking position of a passive object.

A prototype of a 3D camera has been made [4], which uses a single CCD camera to measure “time-of-flight” of modulated infrared light. The output is an “image” of distances, where each pixel is the distance to the object. This 3D camera is very complex and very expensive to realize at present.

1.3 Contributions

The main contributions by the author of this thesis is listed below.

Development of a 3D Reflection Map Model for modeling the intensity of light reflected by a ball, given the position and orientation of the light emitting point source, the light sensitive photodiode and the position of the ball. The 3D Reflection Map Model has been validated by laboratory experiments. Presented in [5].

Development of a model based method to estimate the position of the ball from intensity measurements of light reflected by the ball, using the Nelder-Mead Simplex Algorithm. Results are realization of a 3D input device for a computer, named “non-Touch Screen”, where the input device tracks the 3D position of the users hand in front of the computer screen. Presented in [6]

Investigation on robustness and sensor placement for the non-Touch Screen.

Invention of a human machine interface (HMI) for disabled persons. The HMI is based on non-contact infrared sensors and makes it possible for a person e.g. to control the motion of an electrically powered wheelchair by use of head movements. A prototype of the HMI has been successfully tested on a real electrically powered wheelchair. Patent pending [18]. The HMI is resented in [7] and [9].

Some investigation on infrared sensors for use in robot navigation including modeling of reflections from environment primitives. Preliminary results obtained on distance independent orientation estimation for some of the environment primitives.

1.4 Thesis Outline

The remaining parts of this thesis is organized as follows:

Light Reflection Models introduces the theory of light and reflection of light. A number of different light reflection models are introduced with the purpose of selecting a reflection model that is suited for the construction of a model of how light is reflected from an object.

Sensors introduces the infrared sensors used in the laboratory experiments later in this thesis. The sensor hardware and the signal processing related to the sensors are described.

3D Reflection Map Model derives a model of how light is reflected by a sphere shaped object. Based on the position, orientation and characteristics of one light emitter, one light detector (receiver), the position, size and reflectivity of the spherical object, the intensity of light reaching the receiver, form the emitter, reflected by the object is modeled.

The model is evaluated by comparison to measured light intensities.

3D Position from Reflections develops a method for estimating the three-dimensional position of the spherical object based on (modeled) measurements of reflected light. The estimation procedure developed is based on parameter search in the 3D Reflection Map Model. The method is tested in laboratory experiments.

Realizing the "non-Touch Screen" investigates the robustness and computation time in the 3D reflection from Positions method. Reports results on an implementation of a 3D input device for a computer, by using the 3D Position from Reflections method with the infrared sensors placed along the boarder of a computer screen. The spherical object is successfully replaced by a human hand as pointing device. The sensor setup along the border of the computer screen is also investigated for finding the optimal sensor placement.

2D Position from Reflections develops two methods for estimating 2D positions from measurements of intensities of reflected light. The first method is similar to the 3D Position from Reflections method constrained to two dimensional object positions. The second method is not based on the 3D Reflection Map Model, here the 2D position is found directly from the measured intensities of reflected light.

Realizing a Head Sensor for Controlling a Wheelchair describes the implementation of the second 2D position from Reflections method as human-machine interface (HMI) for control of wheelchair movements by use of head movements. The HMI is implemented and tested on the SIAMO wheelchair, with which test drives has taken place.

Robot Orientation Sensor alters the 3D Reflection Map Model for modeling reflections from environment primitives such as planes, edges, corners and cylinders. The alternated reflection models are intended for use in a robot for navigation purposes. Preliminary results in this area has been obtained using modeled reflections.

Conclusion and Discussion concluding remarks on the work in this thesis, together with some discussion of the results and future work in the field.

Chapter 2

Light Reflection Models

In this chapter the theory of light reflection is presented. A more detailed description of a number of light reflection models are given, with the purpose of selecting an appropriate light reflection model for use in the 3D Reflection Map Model presented in Chapter 4.

2.1 Light and Reflection of Light

The phenomena called light is electromagnetic waves with wavelengths in a the range from approximately 1mm to 0.6nm. Light is further divided into infrared light with wavelength from approx. 1mm to 700nm, visible light in the range 700nm to 400nm, and ultraviolet light with wavelengths in the range approx. from 400nm to 0.6nm.

The light used in this project is infrared light from light emitting diode (LED) these are usually made with wavelengths 880nm or 950nm.

The energy of a light wave is present in bundles of energy called photons. A photon is a package consisting of a certain amount of energy, where the amount of energy depend on the wavelength of the light wave. Variations in light intensity is thus variations in the number of photons. For further theory of light see e.g. [15].

The law of reflection is

$$\theta_{\mathrm{o}} = \theta_{\mathrm{i}} \tag{2.1}$$

where θ_{o} is angle of reflection and θ_{i} is angle of incidence. The law reflection states that the angle of reflection equals the angle of incidence, for a smooth mirror-like surface. Both incidence ray, reflected ray and surface normal is contained in the same plane. This is shown in Figure 2.1.

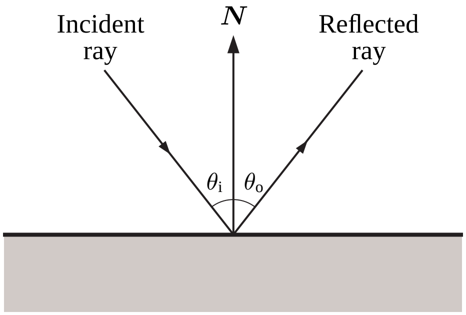


Figure 2.1: Illustration of the law of reflection, the angle of reflection equals the angle of incidence.

When the surface that reflects the light is rough the law of reflection is still valid, but the reflected light ray may be reflected in an other direction than expected from the law of reflection due to the roughness of the surface. This is illustrated in Figure 2.2 where the surface to the left is smooth and the surface to the right is rough. Rough surfaces can be considered as a collection of small smooth surfaces with random orientation, with scattering of the reflected light as a result.

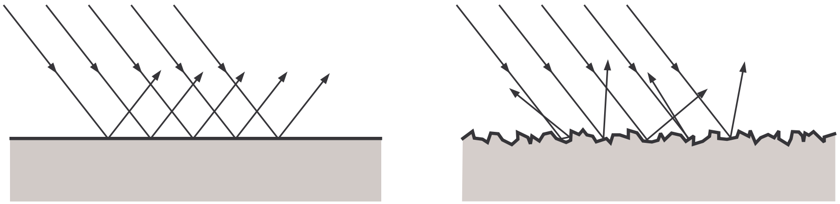


Figure 2.2: Left: Light rays reflected by a smooth surface. Right: Light rays reflected by a rough surface.

The way the light is reflected by a smooth surface is called specular reflection and the way the light rays are reflected (and scattered) by the rough surface is called diffuse reflection. The amount of scattering depends on the roughness of the reflecting surface, several light reflection models have been proposed to model the actual light reflection for various materials, a few of these are briefly reviewed in the following.

2.2 Lambertian Reflection Model

The Lambertian light reflection law is a reflection law for rough surfaces. The assumption is that the reflected light is completely scattered giving a completely diffuse reflection. Thus, the intensity of the reflected light is independent of the angle to the observer.

Lamberts law states that the reflected light intensity L_o is

$$L_\mathrm{o} = \frac{1}{\pi} L_\mathrm{i} \cos\theta_\mathrm{i} \tag{2.2}$$

Where L_i is the intensity of the incoming light, θ_i is the inclination angle.

2.3 Torrance-Sparrow Reflection Model

Torrance and Sparrow proposed a light reflection model in their 1967 paper [30]. This light reflection model has become known as the Torrance-Sparrow reflection model.

The Torrance-Sparrow reflection model takes both specular and diffuse reflection into account, such that a large variety of different reflection properties of different materials can be modeled using this light reflection model.

Figure 2.3 shows the geometry of the setup for the model, with light source, receiver and neighborhood of a point P on the reflecting surface.

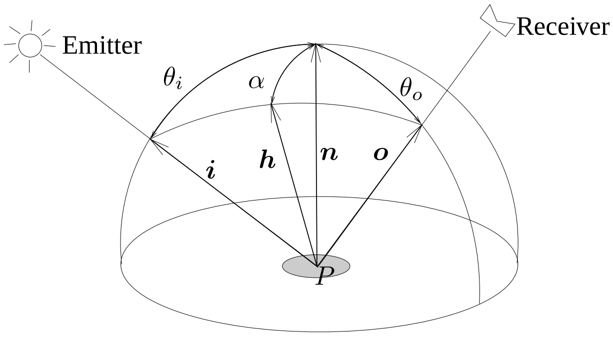


Figure 2.3: The geometry of the Torrance-Sparrow reflection model.

The model can be stated as a sum of diffuse and specular reflection contributions.

$$L_o = \rho_d f_d + \rho_s f_s$$

where $\rho_d + \rho_s = 1$ denotes respectively the diffuse and specular reflection coefficient also called albedo. f_d is the diffuse contribution and f_s is the specular contribution.

The diffuse reflection follows Lamberts law (2.2), which can be rewritten in terms of vectors as

$$f_d = \frac{1}{\pi} L_i \cos\theta_i = \frac{L_i}{\pi} \boldsymbol{i}^\top \boldsymbol{n}$$

where L_i is the incoming light intensity and θ_i is the incidence angle. $\boldsymbol{i}$ and $\boldsymbol{n}$ are direction vectors for the incoming light ray and the surface normal respectively.

The specular part of the reflection from rough surfaces is modeled geometrically by assuming that the rough surface is a collection of micro facet mirrors with orientations that is normal distributed (the mean value coincides with the surface normal). The specular reflection is hence rotationally symmetric.

The micro facet mirror that gives specular reflection in the direction of the receiver have to have their normal vector in the specular direction. The specular direction $\boldsymbol{h}$ is the direction of the bisector of $\boldsymbol{i}$ and $\boldsymbol{o}$ and is found as

$$\boldsymbol{h} = \frac{\boldsymbol{i} + \boldsymbol{o}}{\|\boldsymbol{i} + \boldsymbol{o}\|}$$

The specular reflection in the Torrance-Sparrow model is

$$f_s = L_i \frac{1}{\cos\theta_o} e^{-k\alpha^2} = \frac{L_i}{\boldsymbol{o}^\top \boldsymbol{n}} e^{-k(\arccos \boldsymbol{h}^\top \boldsymbol{n})^2}$$

where θ_o is the angle between surface normal and viewer direction, α is the angle between the specular direction and the surface normal.

The parameters in the Torrance-Sparrow reflection model are ρ_d, ρ_s and α. The albedos ρ_d and ρ_s determines the ratio between diffuse and specular contributions. The parameter α control how narrow the cone of specular reflection is, small α results in a narrow cone and large α results in a wide cone.

Note, that for $\rho_d = 1$ and $\rho_s = 0$ the Torrance-Sparrow model coincides with the Lambertian reflection model, and that ideal mirror reflection results from setting $\rho_d = 0$, $\rho_s = 1$ and $\alpha = 0$.

2.4 Phong's Reflection Model

An other light reflection model often found in the literature is Phong's model, proposed by Phong in his 1975 paper [26]. Phong proposed this model for shading computer graphic objects but the model has also been used to model light reflected by physical objects in e.g. [21] and [25].

Phong wanted a shading model that was fast, and in some way imitated real physical shading situations. The Phong model for calculating the reflected light intensity can be stated as

$$L_\mathrm{o} = \rho \frac{L_\mathrm{i}}{\pi} \big(r_\mathrm{d} \cos(\theta_\mathrm{i}) + (1 - r_\mathrm{d}) \cos^m(\theta_\mathrm{s}) \big) \tag{2.3}$$

where ρ is the surface reflection coefficient, L_i is the incident optical power, r_d is the ratio of incident light that is reflected diffusely and attains values between 0 and 1. θ_i is the incidence angle, and θ_s is the angle between the direction of reflected light and the line of sight, and m is a power which models the specular reflected light for each material.

The geometry of Phong's model is shown in Figure 2.4.

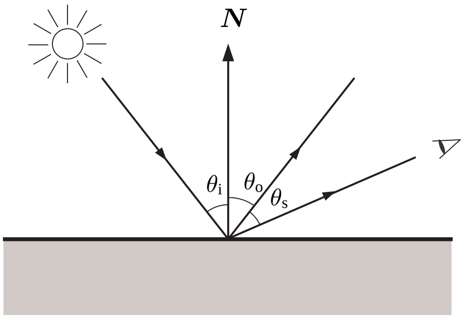


Figure 2.4: The geometry of Phong's reflection model.

2.5 Selection of Reflection Model

The importance of the specular part of the model depends on two things, namely the roughness of the surface of the reflecting object and the beam width (half-power angle) of the emitting light source [21]. Objects with smooth surface has higher specular reflection than rough surfaced objects where the reflection is mainly Lambertian. A reflection model where the light source has a wide beam width, the specular reflection is much less significant than if the light source has a narrow beam width, in comparison to the Lambertian reflection. The border between significance and insignificance of the specular reflection contribution is a half-power angle of approximately 10° when the light source is not too far away from the object [21].

2.6 Summary

Different models of light reflection has been briefly reviewed, and the Lambertian reflection model has been selected as appropriate for the development of the 3D Reflection Map Model in Chapter 4.

Chapter 3

Sensors

This chapter will provide some information on the construction of the infrared sensors used in laboratory experiments described later on in this thesis. The intention is to use low-cost infrared sensors, both emitters and receivers, and that the placement of the sensors should be fixed in the system, such that the object of interest is passive, i.e. no active components are attached to the object.

3.1 Design of Emitter/Receiver pair

As this project is a spin-off from an earlier project called OPTO-control, where similar infrared emitters and receivers also were used, the emitter/receiver pair used in this project is a further development of the one used by Anders la Cour-Harbo in his Ph.D. [16].

The sensors used in this project is infrared sensors, both infrared emitters and infrared receivers. The maximum number of emitter/receiver pairs that can be interfaced to the computer is eight with the hardware available. This sets a limitation on the number of emitters and receivers that can be used for the individual applications described later in this thesis.

Since no active components will be attached to the object of interest the only measurements to be obtained is intensity of reflected light (Methods to measure “time-of-flight” using light exists [4], but is not considered to be a realizable solution for this project). With eight emitter/receiver pairs where the receiver of each pair can only measure the intensity of light reflected light originating from the emitter in the same pair, will result in only eight measurements. This may seem as very few measurements, when considering

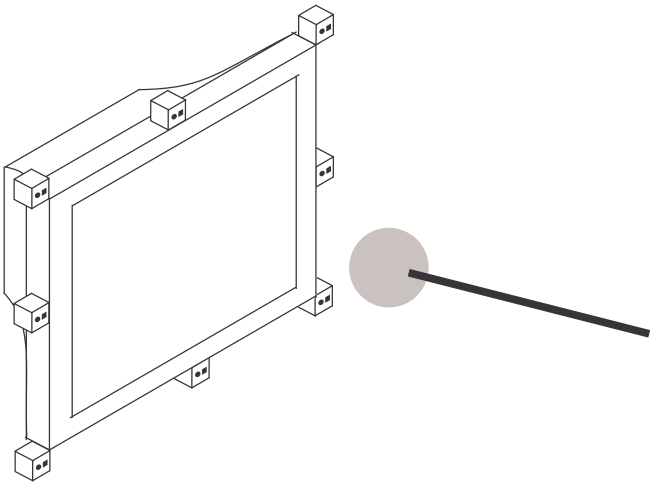


Figure 3.1: Setup for 3D non-Touch Screen with emitter/receiver pairs around the boarder of the computer screen and a ball on a stick for use as pointing device.

the aim of estimating the object position from the intensity measurements of reflected light.

On the other hand with eight emitter/receiver pairs it is possible to obtain 64 dimensional measurements by measuring the intensity of the reflected light originating from each of the eight emitters by all of the eight receivers. The possibility to measure intensity of reflected light from each emitter by each receiver is thus essential in the choice of emitter and receiver for use in this project.

The property of the sensors that each receiver has to be able to measure the intensity of light originating from each individual light emitter is the main reason for the decision to construct a custom infrared emitter and receiver circuit for use in this project. It is also decided to make the emitters and receivers in pairs of one emitter and one receiver. The 3D non-Touch Screen is the first application for which the sensor construction is focused.

At this point a sketch of the expected setup of the non-Touch Screen is useful to help understanding some of the sensor design details, which is shown in Figure 3.1.

With the setup in Figure 3.1 each emitter has to illuminate the entire region of interest in front of the computer screen, in order to have the object reflect light from each emitter towards each receiver. The consequence of this is that the emitters has to have a wide half-angle. The half-angle is the angular deviation from the emitter normal at which the intensity of the emitted light is half of the maximum intensity, which usually is in the normal direction. Also the receiver needs to be sensitive to irradiation from all directions in order to measure the intensity of light from all directions.

The LED SFH487P which has a half-angle of $65°$ is chosen for the emitter, and the photodiode BP104F is chosen for the receiver.

The emitter SFH487P has a peak wavelength of 880nm and a maximum radiance intensity of 30mW/sr, when it is driven by a pulse current of 1A and a pulse time $T_p \leq 100\mu s$. The emitter driver is made as a voltage to current converter, the diagram is shown in Figure 3.2. The circuit is designed to convert an input in the range of 0 to 5V into a current trough the LED in the range of 0 to 1A.

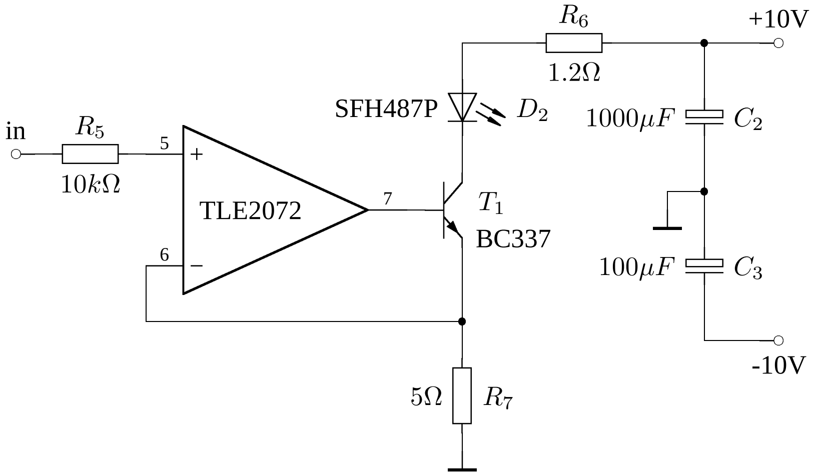


Figure 3.2: Diagram of the emitter part of the sensor.

For the Head Sensor application of Chapter 8, the emitter LED has been changed to two SFH476N connected in series, and angled such that the characteristics of the two LED's together have a half-angle of approx. $60°$ horizontally. SFH476N has a half-angle of $28°$, and a peak wavelength of 830nm with a maximum radiance intensity of 45mW/sr, when it is driven by a pulse current of 1A and a pulse time $T_p \leq 100\mu s$.

The photodiode BP104F chosen for the receiver has a peak sensitivity at wavelength 950nm. This spectral misalignment between emitter and receiver will result in an approximate 50% lower transmitted light power, and thus lower output signal from the receiver. The reason for choosing BP104F as receiver is its relatively low (parasitic) capacitance. The diagram of the receiver circuit is shown in Figure 3.3. The photodiode D_1 generates a current proportional to the amount of irradiated light, which is converted to a voltage by resistor R_3. The (parasitic) capacitance of D_1 of 48pF, together with R_3

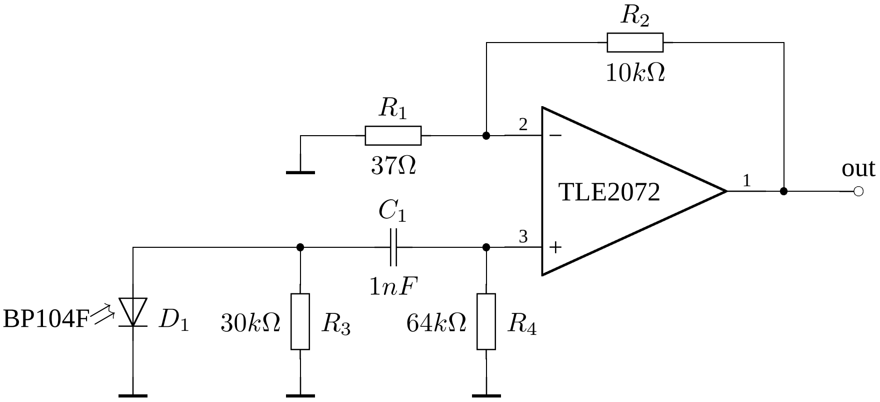


Figure 3.3: Diagram of the receiver part of the sensor.

forms a low-pass filter which has a cut-off frequency of

$$f_0 = \frac{1}{2\pi C_p R_3} = \frac{1}{2\pi \times 48 \cdot 10^{-12} \times 30 \cdot 10^3} = 110.5\text{kHz}$$

which limits the bandwidth of the transmitted signal.

A high-pass filter consisting of C_1 and R_4 is mounted on the input to the amplifier to limit the influence of artificial lighting which causes a 100Hz sine signal overlaying the signal emitted by the emitters. The cut-off frequency of the high-pass filter is chosen to be 2.5KHz.

The op-amp and R_1 and R_2 amplify the signal to a level suitable for sampling into the computer.

The reflecting object is a ball of diameter approx. 60mm and the maximum distance of interest is chosen to be half a meter.

To detect the reflected light from the ball at a distance of half a meter the gain in the receiver amplifier has to be sufficiently large, such that the output of the receiver can be sampled by the computer. On the other hand, the distance from emitter/receiver pair to object at which the receiver will go into saturation increase with larger gain. Tests of amplification levels for the receiver amplifier in the laboratory has shown that a gain of approx. 270 is an appropriate value. With this gain the reflections of a ball at approximately half a meter can be detected, and the distance of saturation is reasonable low with this gain, below 10cm from front of the sensor pair to the center of the ball.

3.2 Sensor Characteristics

One of the aims in this project is to use low-cost standard components such that the applications described later on won't be unattractive because of a high price, due to use of expensive high-precision components. Prior experience with low-cost standard components such as light emitting diode (LED) have shown that the radiation characteristics of the individual samples may differ substantial from the characteristics given in the data sheets. This is the reason for investigating the characteristics of the emitters and receivers used.

3.2.1 Emitter

The characteristic of the emitter SFH487P, chosen for use in the Non-Touch Screen setup of Chapter 5 and Chapter 6, has been measured with the purpose of finding a model of the emitter characteristics.

To measure the radiant characteristic the LED and a receiver have been lined up, and then intensity measurements of the light at the fixed positioned receiver have been made with the emitter rotated in different angles to the receiver. The setup is shown in Figure 3.4.

Figure 3.4: Setup for measuring radiant intensity for the emitters.

The radiant characteristic of eight LED's have been measured for two different rotations of the emitter: $0°$, $90°$. For emitter no. 1 additionally measurements of orientation $45°$ and $-45°$ has been made. The result of these measurements are seen in Figure 3.5, where spline approximations of the radiant characteristics of the individual LED's are shown. The x and y axis are the angles of direction and the z show the intensity of light in that direction.

From Figure 3.5 it is obvious that the radiant characteristic of one sample of SFH487P can differ substantially from the characteristics given in the data-sheet.

To quantify the difference an rotation invariant approximation of the radiant characteristic have been made. Figure 3.6 shows all measurements for $0°$ and $90°$ emitter orien-

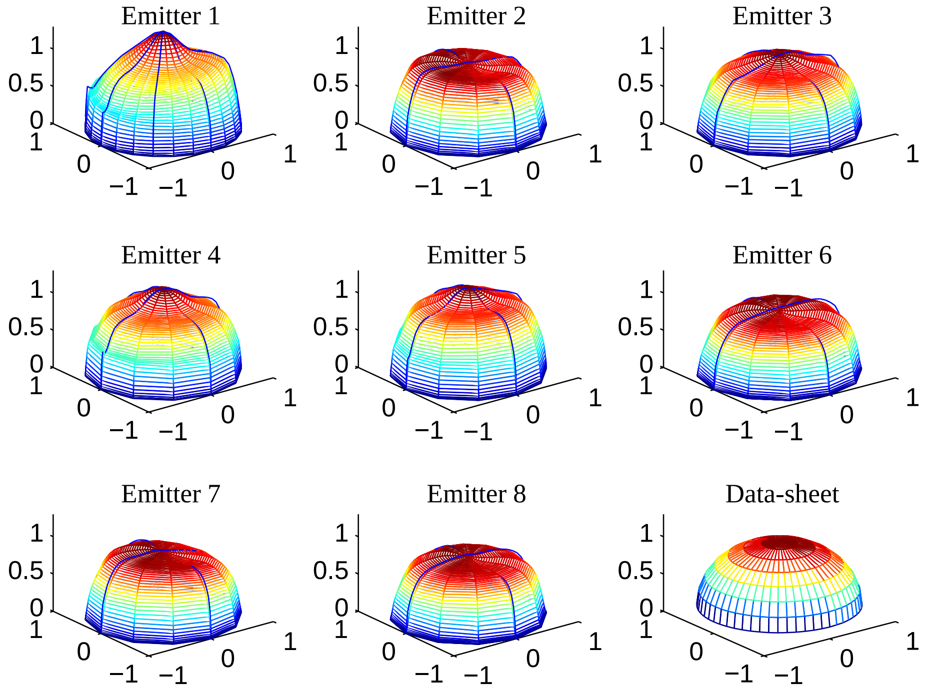


Figure 3.5: Spline approximation of radiant characteristics for eight SFH487P LED's and the data-sheet characteristics.

tation (dotted lines) together with the least square approximation with a dilated cosine $I(\theta) = b\cos(a\theta)$ (solid line), this is close to the data-sheet characteristic ($a = 0.9$), the scalar b is used to compensate various gains in the measurement system (receiver, amplifier, ADC, etc.).

The resulting coefficients of the approximation with a cosine are $a = 0.913$ and $b = 1452$.

The emitter SFH476N is used in the robot setup of Chapter 9 and the "Head sensor" setup of Chapter 8. The data-sheed characteristic shows a nice bell-shaped radiation, but measurements shows that individual samples may deviate substantially from this. Measurements of radiation characteristics for eight samples of SFH476N are shown in Figure 3.7. For the "heas sensor" setup, each emitter is comprised of two SFH476N in series to give more optical power, and mounted with approximately 30 degrees between the normal directions to increase the half angle.

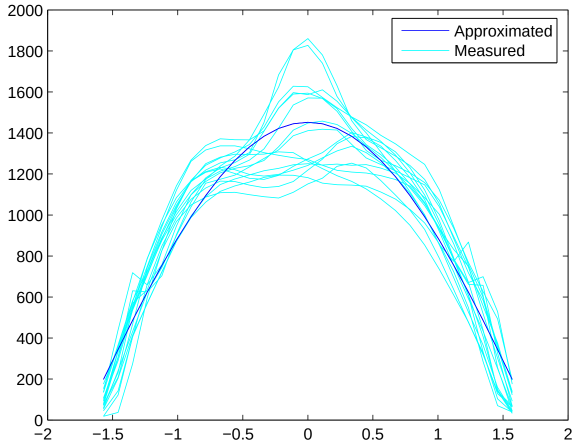


Figure 3.6: Measured and approximated emitter characteristics (x axis is angle in radians).

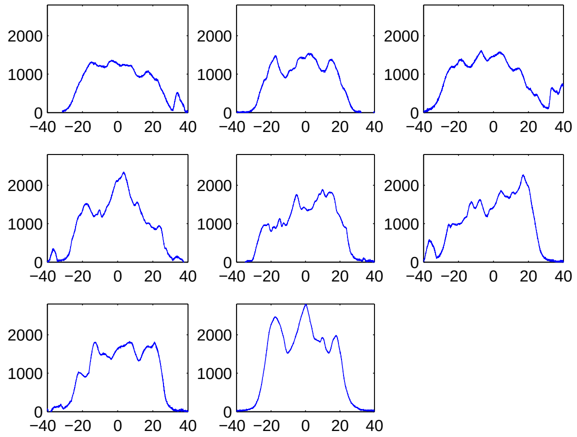


Figure 3.7: Measured characteristics of eight different SFH476N LED (x axis is angle in degrees).

3.2.2 Receiver

The same investigation has been been done for the receiver BP104F as for the emitter. The receivers BP104F is expected to have a characteristic as $I_o(\theta) = a\cos(\theta)$. Figure 3.8 shows the measured receiver characteristics for eight BP104F photodiodes together with a cosine approximation.

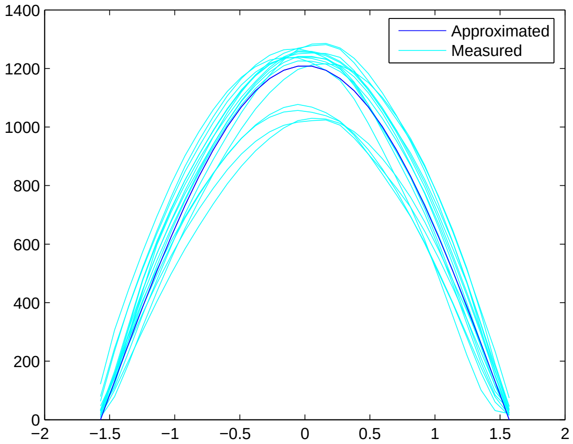


Figure 3.8: Measured and approximated receiver characteristics (x axix is angle in radians).

The characteristics of the receivers are found to be quite close to a cosine, the only evident variation is that there are a difference in the output level from the individual receivers.

3.3 Simultaneous Measurement of Reflections

With eight emitters and eight receivers there it has to be possible to measure the intensity of light on each receiver that originates from each emitter in a “noisy” environment some appropriate modulation or codification of the emitted light signals are needed. The codification serve two purposes, one is to makes it possible to distinguish between light originating from different emitters, and the second is to impose robustness against noise from other light sources e.g. artificial lighting. The disturbances from artificial lighting is usually frequency localized e.g. 100Hz (in Europa) for ordinary light bulb. Newer high frequency low energy light have different frequency of disturbances e.g. 40kHz.

In order to make optical signals from different emitters separable, either time multiplexing of the emitted signals can be used, or the signals emitted by different emitters can be sequences of e.g. orthogonal signals. The last method has the advantage that the individual emitters can emit their respective signals simultaneously.

The solution of emitting sequences of orthogonal sequences from the individual emitters are chosen. Other properties of the sequences in addition to orthogonality can improve the performance of the system, e.g. better noise rejection. For further investigation on this topic see [16], [19], where waveletpackage and Rudin-Shapiro sequences are used for this purpose. General information on wavelets, waveletpackage can be found in e.g. [13], [31] or other books on wavelet theory. Further information on Rudin-Shapiro sequences can be found in [17] or [28].

The symmetric Rudin-Shapiro transform [16], [19] is used here to construct orthogonal spread spectrum signals. The Rudin-Shapiro sequences have the advantage of taking only the discrete values ± 1 and the Rudin-Shapiro transform is it's own inverse. The orthogonality property makes it easy to separate the received mixed signals from each other using inner products.

The length of the Rudin-Shapiro sequences used are 16, this gives eight channels for signals (one channel for each emitter) and eight unused channels, which can be used to estimate the noise in the received signals. The noise estimate from the unused channels can be used to give an estimate of the SNR in the received signal. Further details on this SNR estimation can be found in [16], [19], no further treatment is made in this thesis.

With the 16 bit Rudin-Shapiro sequence modulation used, the hardware is capable of measuring intensity of reflected light on all 64 channels (eight emitters by eight receivers) with a rate of $200Hz$.

3.4 Digital Signal Processing

The signal emitted by the individual emitter is a coded Rudin-Shapiro sequence. The coding is necessary to keep the duty-cycle of the LED's as low as the required 5-6%. The coding is such that one bit in the Rudin-Shapiro sequence is 32 samples, a zero bit is encoded as 32 samples of 0V (LED off), and a one bit is encoded as three samples of 5V (LED on) starting at sample 3, all other samples is 0V (LED off). With this encoding a Rudin-Shapiro sequence of length 16 is emitted as 512 samples. Since the sixteen length 16 Rudin-Shapiro sequences of the Rudin-Shapiro transform each has 10 bit high, the resulting duty-cycle for the LED's are

$$d = \frac{10}{16}\frac{3}{32} = .0586$$

which are within the acceptable level for the emitting LED's.

The sampled signal from Receiver 1 with all eight emitters emitting, is shown in Figure 3.9. Both the low-pass filter and the high-pass filter described in Section 3.1 show their influence on the sampled signal from the receiver. The sampled signal from the receivers has to be decoded such that the Rudin-Shapiro transform can be applied to the received signal.

Figure 3.10 show the first bit coding of the Receiver 1 signal (solid) of Figure 3.9,

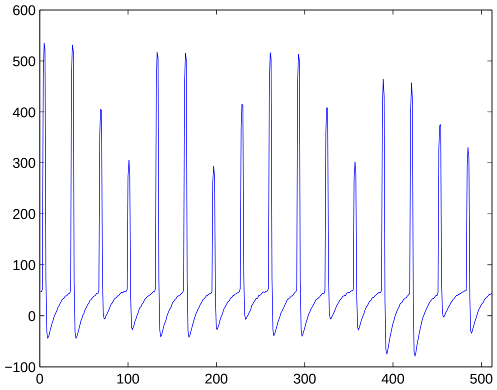


Figure 3.9: Sampled signal from Receiver 1, when all eight emitters are emitting simultaneously.

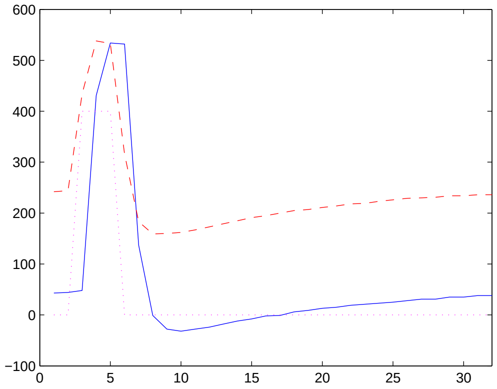


Figure 3.10: Encoding of Bit 1 from sampled signal of Receiver 1 (solid), Receiver 7 (dashed), and encoded output of a high bit (dotted) scaled by 400 to be visible.

together with the first bit coding of the Receiver 7 signal (dashed), and the encoded output of a high bit (dotted) scaled by 400 to be visible.

Two things are obvious from Figure 3.10 the received signals are not equally delayed, and has different offset. The offsets originates from the amplifier circuit, where no attempt has been done to eliminate offset. The x-axis in Figure 3.10 is samples (sample frequency $f_s = 102400$Hz), so the input signals are delayed zero to one sample. This difference in delay is due to multiplexing of the sampling of the input channels in the NI-DAQ6071E sampling card used. The multiplexing causes the first channel to be sampled first, then the second channel and so on. This results in a different time delay for each input channel. The delay for the individual channel are constant, since the output from the used sampling card NI-DAQ6713 are synchronized.

The decoding of the sampled signals from the receivers are

$$\mathrm{RST}_{\mathrm{in}}(k) = \frac{\mathrm{IN}(k*32+3+(j<2)) + \mathrm{IN}(k*32+4+(j<2))}{2} - \mathrm{IN}(k*32)$$

where k are bit number of the Rudin-Shapiro sequence, and j are input channel number $(0,\ldots,7)$, IN is the sampled input signal from the receiver. This also eliminates the offset from the amplifier.

The decoding could be interpreted as a filter

$$\begin{aligned} y(n) &= \tfrac{1}{2}x(n) + \tfrac{1}{2}x(n-1) - x(n-5) && \text{for Receiver 1 and 2} \\ y(n) &= \tfrac{1}{2}x(n) + \tfrac{1}{2}x(n-1) - x(n-4) && \text{for Receiver 3 to 8} \end{aligned}$$

followed by a down-sampling by 32. Figure 3.11 show the frequency response for both filters before down-sampling, filter for Receiver 1 and 2 in the upper plot, and filter for receiver 3 to 8 in the lower plot.

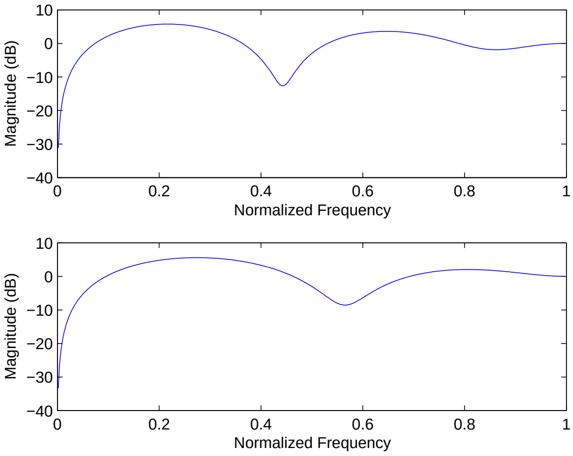


Figure 3.11: Frequency response of decoding filters before down-sampling. Top: Filter for Receiver 1 and 2. Bottom: filter for receiver 3 to 8.

Down-sampling results in “folding” the frequency axis. For the two filters considered here, down-sampling by 32 results in a totally flat frequency response. This is shown in Figure 3.12, where again upper plot show response from filter for Receiver 1 and 2, and lower plot show response from filter for Receiver 3 to 8.

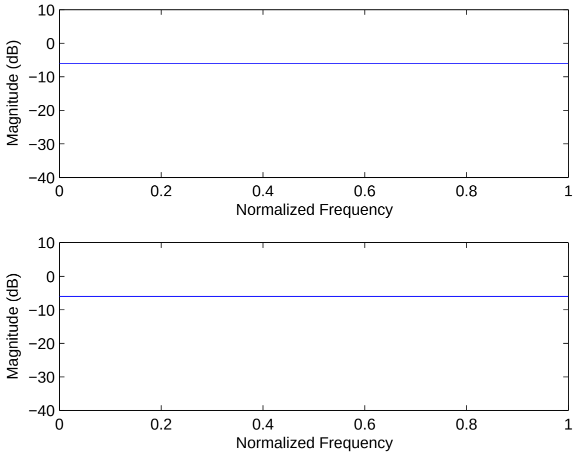


Figure 3.12: Frequency response of decoding filters after down-sampling. Top: Filter for Receiver 1 and 2. Bottom: filter for receiver 3 to 8.

3.5 Noise

There are several sources for noise to enter into the sensors discussed here: Sun light, artificial lighting, light and electromagnetic noise from computer monitors etc. In this section the noise within the receiver circuit is considered.

To measure the noise from the receiver, a series of 10000 measurements are made with emitters turned off, no artificial light and only the absolute necessary electrical equipment turned on. Figure 3.13 show a spectrogram of the first 10000 samples from the signal sampled from Receiver 1.

There are some obvious patterns in the spectrogram in Figure 3.13, indicating that the noise on the sampled input is unlikely to be normally distributed. Figure 3.14 show the spectrogram of the decoded and RST transformed data. As seen from the figure, there are no obvious patterns in the spectrogram for the decoded and RST transformed data, indicating that these might be normally distributed.

A formal test for normality, the Jarque–Bera parametric hypothesis test of composite normality (implemented as `jbtest` in the Matlab Statistics Toolbox) is applied to the 10000 samples used for the spectrogram. The test results in rejection of the hypothesis that the noise on the sampled signal is normally distributed, at a significance level of 0.05. Applying the same formal test to the decoded and RST transformed data, gives the result that the hypothesis can not be rejected at a significance level of 0.05.

A second series of 100000 measurements are obtained for verification on a larger set of

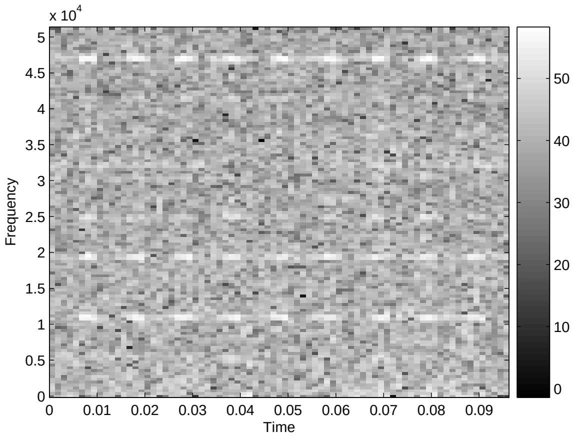


Figure 3.13: Spectrogram of sampled signal from Receiver 1, with all emitters turned off.

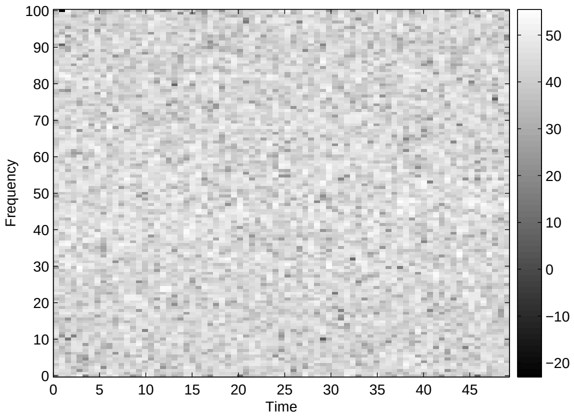


Figure 3.14: Spectrogram of decoded and RST transformed data from Receiver 1, with all emitters turned off.

data. Applying the Jarque–Bera test to the 100000 decoded and RST transformed data in this set gives the same result as before: The hypothesis that they are normally distributed can not be rejected at a significance level of 0.05.

3.6 Summary

This chapter has been devoted to describing the sensors used in the laboratory experiments during this project work. The emitter and receiver circuits were described, and the characteristics of the emitters and receivers was measured. The measured characteristics of the emitters was found to be quite different for the individual emitters, and in general deviate from the (mean) characteristic found in the data sheet. A method for simultaneous measurement of reflections was outlined, and the signal processing used in the measurements was also described. Finally the noise on the measurements was showed to be Gaussian.

Chapter 4

3D Reflection Map Model

In this chapter a 3D Reflection Map Model is developed for estimating the intensity of the light that reaches the receiver after being emitted by an emitter and reflected by an object. The model is supposed to give an understanding of how the intensity of light at each receiver in a system with light emitters, light receivers and one object behaves. This understanding will later be used to find the position of the object given the intensities of the reflected light with the purpose of making a "non-Touch Screen", as described in Chapter 5. The intention is to use a human hand as pointing device for the non-Touch Screen, but to make things as easy as possible the human hand is modeled as a ball, which has the nice property of being rotationally invariant. The human arm usually attached to the human hand is neglected in the model.

4.1 Idea of Ray Tracing

Ray tracing is an illumination method used in computer graphics scene rendering. It traces rays from the position of the eye trough the image plane and into the scene. When ever the ray intersects an object in the scene the ray is reflected or transformed (in case of transparent object). The reflected and/or transformed rays are then traced further into the scene intersecting other objects or the light sources. In this way shadows, multiple specular reflections and texture mapping can be handled. Figure 4.1 shows the principle in ray tracing.

For the problem at hand there are no need for handling shadows, multiple specular reflections and texture mapping. First of all because there are only one object in the scene, second because the light from the individual light sources can be encoded such that they are separable at the receivers, for the third the "image" is formed by a photodiode which

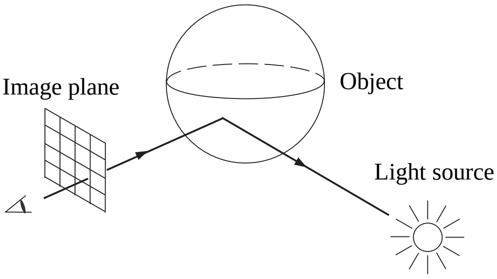


Figure 4.1: The principle in ray tracing.

is comparable to making a one pixel picture. Therefore a simple form of ray tracing can be applied, where there is only one light source (an infrared LED), one object (the hand modeled by a ball), and a one pixel image plane.

Further more it is assumed that there is no line of sight from receiver to emitter. The reason for this is that in a real implementation the intensity of direct light from the emitter onto the receiver will be much higher than the intensity of the light reflected by the object. Since the receiver has to be quite sensitive to measure the intensity of the reflected light, a direct line of sight from receiver to emitter would cause saturation of the receiver, making it impossible to measure the intensity of the light reflected by the object.

A model of light intensity will be developed and evaluated in the remaining part of this chapter. The model consists of one emitter (light source) emitting light into a 3-dimensional space containing a sphere (object) which reflects the light, and one receiver which measures the intensity of light reaching it. The output of the model is the intensity of the light that reaches the receiver after being reflected by the sphere (object), given the position and direction of the emitter and the receiver and the position of the sphere. The model will not include direct line of sight from emitter to receiver.

4.2 Reflection Model

The emitter, the receiver and the center of the object will always lie in a plane. For simplification assume that the emitter is placed in origo and the receiver is placed on the positive part of the y axis and the object (sphere) has positive x coordinate, this can always be achieved by translation and rotation.

4.2.1 Identification of the Emitter/Reciever/Center Plane

The coordinate system used in the model has the emitter in origo, the receiver on the positive y axis and the sphere center in the part of the xy-plane with positive x coordinate.

To find this coordinate system from the position of the emitter, $\mathcal{E}$, the position of the receiver, $\mathcal{R}$, and sphere center, $\tilde{C}$, translation and rotation are used. First translation is used to place the emitter in origo:

$$\boldsymbol{x}_{\mathcal{E}} = \boldsymbol{x} - \mathcal{E}$$

Then rotation around the z axis is used to place the receiver on the y axis, this is obtained by the rotation matrix:

$$T = \begin{bmatrix} \frac{\mathcal{R}_y - \mathcal{E}_y}{\|\mathcal{E}-\mathcal{R}\|} & -\frac{\mathcal{R}_x - \mathcal{E}_x}{\|\mathcal{E}-\mathcal{R}\|} & 0 \\ \frac{\mathcal{R}_x - \mathcal{E}_x}{\|\mathcal{E}-\mathcal{R}\|} & \frac{\mathcal{R}_y - \mathcal{E}_y}{\|\mathcal{E}-\mathcal{R}\|} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Let $\tilde{\boldsymbol{x}}$ denote the coordinates of $T\boldsymbol{x}_E = T(\boldsymbol{x} - E)$. Finally a rotation around the y axis so that the center of the sphere has z coordinate zero is applied

$$T_1 = \begin{bmatrix} \frac{\tilde{C}_x}{\sqrt{\tilde{C}_x^2+\tilde{C}_z^2}} & 0 & \frac{\tilde{C}_z}{\sqrt{\tilde{C}_x^2+\tilde{C}_z^2}} \\ 0 & 1 & 0 \\ -\frac{\tilde{C}_z}{\sqrt{\tilde{C}_x^2+\tilde{C}_z^2}} & 0 & \frac{\tilde{C}_x}{\sqrt{\tilde{C}_x^2+\tilde{C}_z^2}} \end{bmatrix}$$

Now the coordinates $\boldsymbol{x}'$ which is used to find the reflection for the individual emitter/receiver pair is found as

$$\boldsymbol{x}' = T_1 T(\boldsymbol{x} - \mathcal{E})$$

In the $\boldsymbol{x}'$ coordinate system the direction of a particular emitter (receiver) obviously change when considered together with different receivers (emitters). A unit normal vector of the direction of each emitter and receiver denoted $\boldsymbol{n}_{\mathcal{E}}$ respectively $\boldsymbol{n}_{\mathcal{R}}$ are used to keep track of these directions.

4.2.2 The Model of Light Reflected by a Ball

The model of reflected light is a geometric model based on the assumption that the light source continuously emits a constant number of photons per time unit. The light source is assumed to be a point source.

The light is propagating in all directions from the light source. It is assumed that the intensity of the light degrades with the square of the distance to the light source, such that the integral

$$\int_S L_{o,source}(\phi, \theta)\, dA = const.$$

where S is a spheres with center at the light source and arbitrary radius and surface element dA. ϕ and θ are angles that describe the light ray direction.

In the following consider one light source, one receiver and one ball, and assume no light travels directly from light source to receiver. The light source is assumed to have an intensity characteristic denoted by $L_i(\nu, \theta, \boldsymbol{n}_{\mathcal{E}})$ where ν and θ are angles of direction with respect to $\boldsymbol{n}_{\mathcal{E}}$.

When the ball reflects light it can be considered as a light source with intensity depending on the intensity and direction of the light it reflects. The intensity of the light reaching the receiver from a single point on the ball can then be found as

$$I = \int_{\mathcal{R}} L_o(P, \phi, \rho, \eta, \boldsymbol{n}_r)\, d\rho\, d\eta$$

where $\boldsymbol{n}_r$ is the normal vector of the receiver, ϕ is the incidence angle at the sphere, ρ and η is the angles of direction to the receiver from the point P on the ball.

Integrating over all the points in sight of both light source and the receiver gives the following equation for the intensity of the light reaching the receiver.

$$I = \int_{\mathcal{P}} L_i(\nu, \theta, \boldsymbol{n}_{\mathcal{E}}) \int_{\mathcal{R}} L_o(P, \phi, \rho, \eta, \boldsymbol{n}_r)\, d\rho\, d\eta\, d\theta\, d\nu \tag{4.1}$$

where $\mathcal{P}$ is the area on the ball that is in line of sight of both light source and receiver.

The integration limits can be found from the position of ball center and the position and direction of the light source and receiver, as described in the next sections.

4.2.3 Determining the Integration Limits for the Model

The area on the sphere visible to the emitter or the receiver is a small circle on the sphere. The area visible by both the emitter and receiver is the intersection of two small circles as shown on figure 4.2.

Note, that the visible area is symmetric around the xy plane. This symmetry is only exploited when determining the area visible to both emitter and receiver. The symmetry can not be exploited when evaluating the model because this would require that the normal direction of both the emitter and the receiver is contained in the xy plane.

To find the point Q the tangent points of the emitter and receiver cones on the circle is needed, see figure 4.3. To find these points solve an equation with a line trough the emitter (receiver) has distance R to the center of the circle.

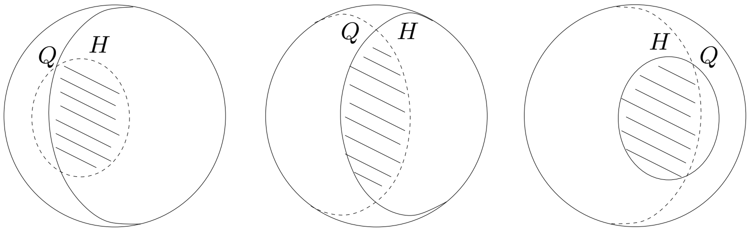


Figure 4.2: The three possibilities for areas visible to both the emitter and the receiver. Receiver: Dashed line, Emitter: Solid line.

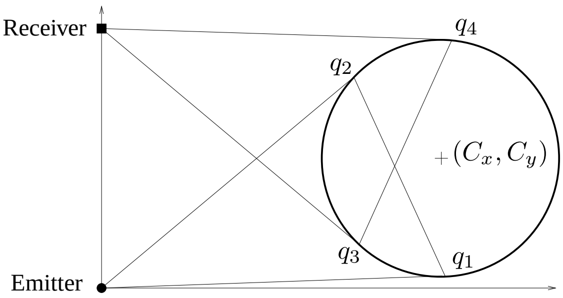


Figure 4.3: The four points necessary to find the point Q.

First the slopes of the lines from the emitter can be found by solving an equation where a line trough origo has distance R to the point (C_x, C_y). This can be obtained by

$$\frac{\left|\begin{bmatrix}1\\ \alpha\\ 0\end{bmatrix} \times \begin{bmatrix}C_x\\ C_y\\ 0\end{bmatrix}\right|}{\sqrt{1+\alpha^2}} = R \qquad \Leftrightarrow \qquad (C_y - C_x\alpha)^2 = R^2 + R^2\alpha^2$$

when rewritten

$$(C_x^2 - R^2)\alpha^2 - 2C_xC_y\alpha + C_y^2 - R^2 = 0$$

which has solutions

$$\begin{aligned}\alpha_1 &= \frac{C_xC_y - R\sqrt{C_x^2 + C_y^2 - R^2}}{C_x^2 - R^2}\\ \alpha_2 &= \frac{C_xC_y + R\sqrt{C_x^2 + C_y^2 - R^2}}{C_x^2 - R^2}\end{aligned} \tag{4.2}$$

For the lines trough the receiver a similar derivation gives

$$\alpha_3 = \frac{C_x(C_y - R_r) - R\sqrt{C_x^2 + (C_y - R_r)^2 - R^2}}{C_x^2 - R^2}$$
$$\alpha_4 = \frac{C_x(C_y - R_r) + R\sqrt{C_x^2 + (C_y - R_r)^2 - R^2}}{C_x^2 - R^2}$$

Remark: This disregards a small area where only a part of the receiver is 'visible'.

To find the tangent points on the circle find intersection of the line trough the emitter respective the receiver with the slopes found above and the circle, the equations are for the emitter ($i = 1, 2$)

$$t(1, \alpha_i)\ , \qquad\qquad (x - C_x)^2 + (y - C_y)^2 = R^2$$

This is determined by the t that solves

$$(1 + \alpha_i^2)t^2 - 2(C_x + \alpha_i C_y)t + C_x^2 + C_y^2 = R^2$$

This has solution

$$t = \frac{C_x + \alpha_i C_y \pm \sqrt{(R^2 - C_x^2)\alpha_i^2 + 2\alpha_i C_x C_y - C_y^2 + R^2}}{1 + \alpha_i^2} = \frac{C_x + \alpha_i C_y}{1 + \alpha_i^2}$$

since it is a tangent point the square root is zero. Inserted into the line gives the points q_1 and q_2. The coordinates are denoted x_i and y_i.

For the receiver the lines intersecting the circle are ($i = 3, 4$)

$$t(1, \alpha_i) + (0, R_r) \tag{4.3}$$

which is solved similar to the above and gives the points q_3 and q_4.

The intersection of these lines gives the x and y coordinates of Q. This is

$$Q_x = \frac{\begin{vmatrix} x_1 - \frac{x_2 - x_1}{y_2 - y_1} y_1 & \frac{x_1 - x_2}{y_2 - y_1} \\ x_3 - \frac{x_4 - x_3}{y_4 - y_3} y_3 & \frac{x_3 - x_4}{y_4 - y_3} \end{vmatrix}}{\begin{vmatrix} 1 & \frac{x_1 - x_2}{y_2 - y_1} \\ 1 & \frac{x_3 - x_4}{y_4 - y_3} \end{vmatrix}} \qquad Q_y = \frac{\begin{vmatrix} 1 & x_1 - \frac{x_2 - x_1}{y_2 - y_1} y_1 \\ 1 & x_3 - \frac{x_4 - x_3}{y_4 - y_3} y_3 \end{vmatrix}}{\begin{vmatrix} 1 & \frac{x_1 - x_2}{y_2 - y_1} \\ 1 & \frac{x_3 - x_4}{y_4 - y_3} \end{vmatrix}}$$

The z coordinate is then found from the sphere equation.

To find the heighest point H on the sphere visible to both emitter and receiver, the parameterization of the lines q_2q_1 and q_3q_4 are considered, since the placement of the point (Q_x, Q_y) on this line gives information on which of the tree possibel situations from figure 4.2 is present.

First the parameter values for the point (Q_x, Q_y) is found:

$$f_m = \frac{\|q_3 - (Q_x, Q_y)\|}{\|q_3 - q_4\|}$$
$$f_s = \frac{\|q_2 - (Q_x, Q_y)\|}{\|q_2 - q_1\|}$$

Note, that f_m and f_s can not both be greater than $\frac{1}{2}$ since the area visible to both emitter and receiver is less than half the sphere.

When $f_m \geq \frac{1}{2}$ the point with highest z coordinate is the point on the receiver cone tangent line with highest z coordinate, figure 4.2 right.

$$H_x = \frac{1}{2}(x_4 - x_3) + x_3$$
$$H_y = \frac{1}{2}(y_4 - y_3) + y_3$$
$$H_z = \sqrt{R^2 - (H_x - C_x)^2 - (H_y - C_y)^2}$$

When $f_s \geq \frac{1}{2}$ the point with highest z coordinate is the point on the emitter cone tangent line with highest z coordinate, figure 4.2 left.

$$H_x = \frac{1}{2}(x_1 - x_2) + x_2$$
$$H_y = \frac{1}{2}(y_1 - y_2) + y_2$$
$$H_z = \sqrt{R^2 - (H_x - C_x)^2 - (H_y - C_y)^2}$$

When $f_m < \frac{1}{2}$ and $f_s < \frac{1}{2}$ the point with highest z coordinate is Q.

$$H_x = Q_x, \quad H_y = Q_y, \quad H_z = Q_z$$

The minimum angle ν_1 and maximum angle ν_2 of a plane that intersects the sphere (as shown on figure 4.4) is

$$\nu_2 = \arcsin \frac{H_z}{\|H\|}\ , \qquad\qquad \nu_1 = -\nu_2$$

Remark: A small area near the boundary that is visible to only a part of the receiver need more attention and maybe some correction!

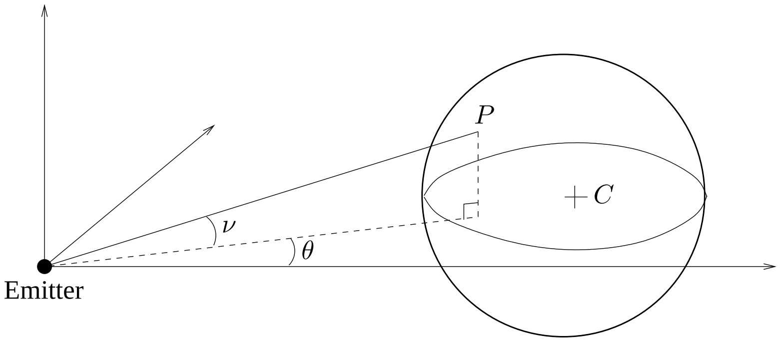


Figure 4.4: The angle ν of planes trough both emitter and receiver intersecting the sphere.

To determine the minimum and maximum θ angles the intersection points between the small circles on the sphere and a plane which contains the y axis and has angle ν with the xy plane.

The small circles are parameterized as

$$\begin{bmatrix} ta + x_1 \\ tb + y_1 \\ \sqrt{R^2 - (ta+c)^2 - (tb+d)^2} \end{bmatrix}, \quad t \in [0,1]$$

where $a = x_2 - x_1$, $b = y_2 - y_1$, $c = x_1 - C_x$, $d = y_1 - C_y$ and where (x_1, y_1) are the parameterization starting point and (x_2, y_2) are the parameterization end point i.e. the start and end points for the emitter small circle are respectively q_2, q_1 and for the receiver small circle q_3, q_4.

The plane which contains the y axis and has angle ν with the xy plane is identified as $\{\boldsymbol{x} \in \mathbb{R}^3 | (x,y,z) = (x,y,x\tan\nu);\ x,y \in \mathbb{R}\}$.

The parameter values for the intersection points are determined by

$$(ta + x_1)\tan\nu = \sqrt{R^2 - (ta+c)^2 - (tb+d)^2}$$

which has the solutions

$$t_1 = \frac{-h - \sqrt{h^2 - g(x_1^2 \tan^2\nu - R^2 + c^2 + d^2)}}{g}$$
$$t_2 = \frac{-h + \sqrt{h^2 - g(x_1^2 \tan^2\nu - R^2 + c^2 + d^2)}}{g}$$

where $h = ax_1 \tan^2\nu + ac + bd$ and $g = a^2\tan^2\nu + a^2 + b^2$.

Now naming parameter values for the intersections between the plane and the emitter small circle t_{e1}, t_{e2} and t_{r1}, t_{r2} for the receiver small circle. Then θ_1 and θ_2 can be found as

$$\theta_1 = \begin{cases} \arcsin \frac{t_{r1}(q_{4y}-q_{3y})+q_{3y}}{\sqrt{(t_{r1}(q_{4x}-q_{3x})+q_{3x})^2+(t_{r1}(q_{4y}-q_{3y})+q_{3y})^2}}, & t_{r1} < f_m \\ \arcsin \frac{t_{e2}(q_{1y}-q_{2y})+q_{2y}}{\sqrt{(t_{e2}(q_{1x}-q_{2x})+q_{2x})^2+(t_{e2}(q_{1y}-q_{2y})+q_{2y})^2}}, & t_{e2} < f_s \end{cases}$$

$$\theta_2 = \begin{cases} \arcsin \frac{t_{r2}(q_{4y}-q_{3y})+q_{3y}}{\sqrt{(t_{r2}(q_{4x}-q_{3x})+q_{3x})^2+(t_{r2}(q_{4y}-q_{3y})+q_{3y})^2}}, & t_{r2} < f_m \\ \arcsin \frac{t_{e1}(q_{1y}-q_{2y})+q_{2y}}{\sqrt{(t_{e1}(q_{1x}-q_{2x})+q_{2x})^2+(t_{e1}(q_{1y}-q_{2y})+q_{2y})^2}}, & t_{e1} < f_s \end{cases}$$

The point P on the sphere can be found by solving an equation where the line from origo with angle ν to the xy plane and angle θ to the x axis has distance R to the center of the sphere. Naming $\alpha = \tan\theta$ and $\beta = \tan\nu$ this gives

$$\left\| \begin{bmatrix} t \\ t\alpha \\ t\beta\sqrt{1+\alpha^2} \end{bmatrix} - \begin{bmatrix} C_x \\ C_y \\ 0 \end{bmatrix} \right\| = R \quad \Leftrightarrow \quad (t-C_x)^2 + (t\alpha - C_y)^2 + (1+\alpha^2)\beta^2 t^2 = R^2$$

The solution of interest is the one closest to origo, which is

$$t_p = \frac{C_x + \alpha C_y - \sqrt{(C_x+\alpha C_y)^2 - (1+\beta^2)(1+\alpha^2)(C_x^2 + C_y^2 - R^2)}}{(1+\beta^2)(1+\alpha^2)}$$

This gives the point

$$P = (t_p, t_p\alpha, t_p\beta\sqrt{1+\alpha^2})$$

on the sphere. The unit normal vector $\boldsymbol{n}_P$ to the sphere at point P is

$$\boldsymbol{n}_P = \frac{P - C}{r}$$

The angle of incidence on the sphere is

$$\phi = \arccos \frac{-P^\top \boldsymbol{n}_P}{\|P\|} \tag{4.4}$$

where $\boldsymbol{n}_p$ is the outer unit normal vector of the sphere at point P.

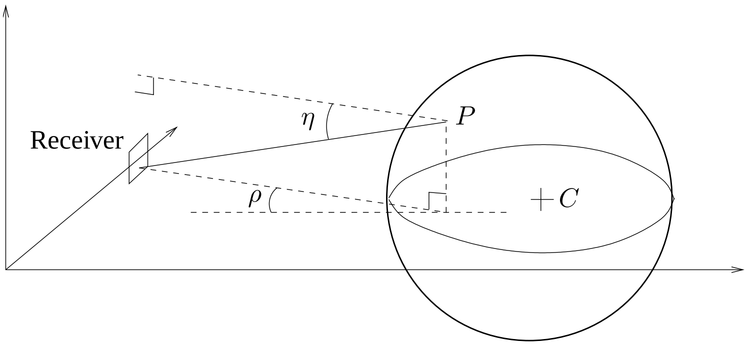


Figure 4.5: The angles η and ρ of a line from the point P on the sphere to the receiver.

The angles of light reflected by the sphere towards the receiver is found as

$$\rho_1 = \arcsin \frac{R_l - P_y}{\sqrt{P_x^2 + (R_l - P_y)^2}}$$

$$\rho_2 = \arcsin \frac{R_r - P_y}{\sqrt{P_x^2 + (R_r - P_y)^2}}$$

$$\eta_1 = \arcsin \frac{R_b - P_z}{\sqrt{P_x^2(1 + \tan^2 \rho) + (R_b - P_z)^2}} \tag{4.5}$$

$$\eta_2 = \arcsin \frac{R_t - P_z}{\sqrt{P_x^2(1 + \tan^2 \rho) + (R_t - P_z)^2}} \tag{4.6}$$

A line from P with angle ρ to a plane trough P parallel to the xz plane and with angle η to a horizontal plane trough P intersects the yz plane in

$$R_\rho = \begin{bmatrix} 0 \\ P_x \tan \rho + P_y \\ P_x \sqrt{1 + \tan^2 \rho} \tan \eta + P_z \end{bmatrix}$$

The receiver characteristic is $\cos \zeta$ where ζ is the angle of incidence, which is found as

$$\cos \zeta = \frac{(P - R_\rho)^\top \boldsymbol{n}_r}{\|P - R_\rho\|} \tag{4.7}$$

4.2.4 Evaluation of the Model Integral

As discussed in the reflection section, the significance of the specular reflection part of the Torrance-Sparrow reflection model is insignificant when the light source has a half-power angle of above $10°$ [21]. The model is to be used for modeling a setup where the light sources has 65° half-power angle, so the reflection model is taken to be pure Lambertian.

With this choice of light reflection the integral equation (4.1) for the model reduces to

$$I = \frac{\rho_d}{\pi} \int_{\nu_1}^{\nu_2} \int_{\theta_1}^{\theta_2} L_i(\nu, \theta, \boldsymbol{n}_{\mathcal{E}}) \cos\phi \int_{\rho_1}^{\rho_2} \int_{\eta_1}^{\eta_2} \cos\zeta \, d\eta \, d\rho \, d\theta \, d\nu \tag{4.8}$$

where L_i is the emitter characteristic, $\cos\phi$ is the reflection model for the sphere, given by (4.4), and $\cos\zeta$ is the receiver characteristic, given by (4.7).

The emitter characteristic was estimated in Section 3.2. The emitter characteristic is implemented as

$$L_i(\nu, \theta, \boldsymbol{n}_{\mathcal{E}}) = \max\Big\{ 0, \frac{P^\top \boldsymbol{n}_{\mathcal{E}}}{\|P\|} \Big\}$$

which differs from the estimated one by not scaling the angle, but takes into account that no light is emitted from the rear side of the emitter, and it is continuous over all directions.

Assuming that ρ_1 and ρ_2 are so close that ρ in (4.5) and (4.6) can be considered constant, it is possible to analytically evaluate the inner double integral and obtain

$$\begin{aligned} \int_{\rho_1}^{\rho_2} \int_{\eta_1}^{\eta_2} \cos\zeta \, d\rho = e_2 \left(\frac{n_{rx} r_2 + n_{ry}}{f_2 g_2} - \frac{n_{ry} + r_1 n_{rx}}{f_2 g_1} \right) + \\ e_1 \left(\frac{r_1 n_{rx} + n_{ry}}{f_1 g_1} - \frac{n_{ry} + r_2 n_{rx}}{f_1 g_2} \right) + n_{rz} \left(\frac{\rho_1 - \rho_2}{f_1} + \frac{\rho_2 - \rho_1}{f_2} \right) \end{aligned}$$

where $e_i = \tan\eta_i$, $r_i = \tan\rho_1$, $f_i = \sqrt{1 + e_i^2}$ and $g_i = \sqrt{1 + r_i^2}$ for $i = 1, 2$.

Search for an analytic expression for the evaluation of the outer integral in (4.8) has not been successful due to the complicated integrand. The outer integral is evaluated numerically.

4.3 Model Validation

The physical setup for validating the model consists of two infrared sensor pairs each with one emitter and one receiver, and a positioning device for moving the ball, see Figure 4.6.

Listed in Table 4.1 are the positions and directions of the sensors used to validate the model.

The reflections from a ball is measured in a grid in front of the sensors. The model adaption consists of multiplying the model by a constant to compensate for various

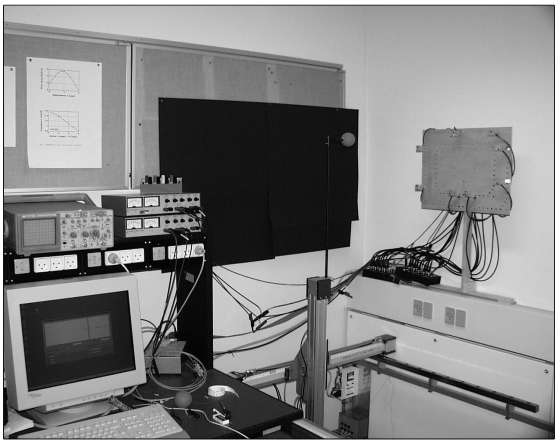


Figure 4.6: The physical setup for the model validation.

	Position (cm)	Normal
Emitter 1	$(0.65, 0, -15.5)$	$(0, 0.707, 0.707)$
Receiver 1	$(-0.65, 0, -15.5)$	$(0, 0.707, 0.707)$
Receiver 2	$(18.3, 0, -15.4)$	$(-0.557, 0.707, 0.435)$

Table 4.1: The sensor placement for model validation measurements.

gains in the physical setup, which are not included in the model e.g. amplification in the receiver to get a signal level suitable for sampling by the computer.

The value of the constant s is obtained by least squares as

$$s = \min_{r \in \mathbb{R}} \sum_{c \in X} (I_m(c, E, R) - rI(c, E, R))^2 \tag{4.9}$$

where X is the 3D grid of $27 \times 15 \times 21$ equidistantly spaced points in the range $[-32.5\text{cm}, 32.5\text{cm}] \times [5\text{cm}, 40\text{cm}] \times [-25\text{cm}, 25\text{cm}]$ of measured ball reflections. $I_m(c)$ is the measured reflections and I_{sat} is the level where the receiver saturates.

The resulting model compared to the measured data is shown in Figure 4.7 and Figure 4.8. Where four of the 21 measured planes for ball center are shown for each of the measurements: Emitter 1 to Receiver 1 and Emitter 1 to Receiver 2 respectively.

The plots shows the reflection of a ball with center in a plane $z = -10, 0, 10, 20$ both measured (surface) and modeled (mesh). The z axis is the measured intensity of the reflected light.

As Figure 4.7 and Figure 4.8 indicates are the modeled reflections close to the measured

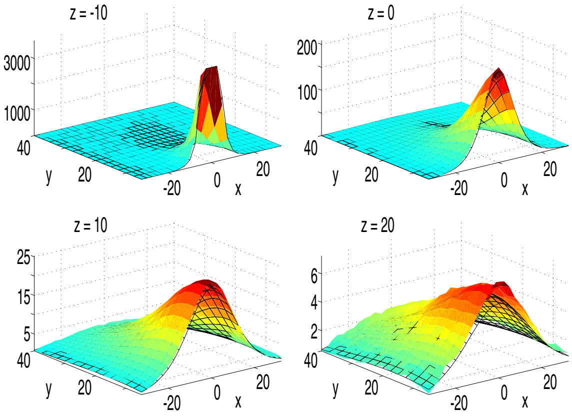


Figure 4.7: Slices from measured (surface) and modeled (mesh) reflections, from Emitter 1 to Receiver 1.

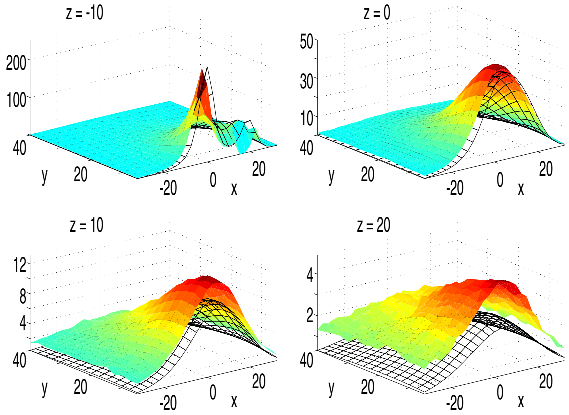


Figure 4.8: Slices from measured (surface) and modeled (mesh) reflections, from Emitter 1 to Receiver 2.

reflections. The apparently larger difference where the ball position has a z coordinate of 20 is due to different scales in the sub-plots of both figures, and reflections from the position device which become significant for ball positions is more than approximately 14cm above the sensor and closer to the sensor plane than approximately 10cm. To get rid of this reflection from the positioning device, the positioning device have to be modified.

The slight misalignment between the modeled and measured reflections when emitter

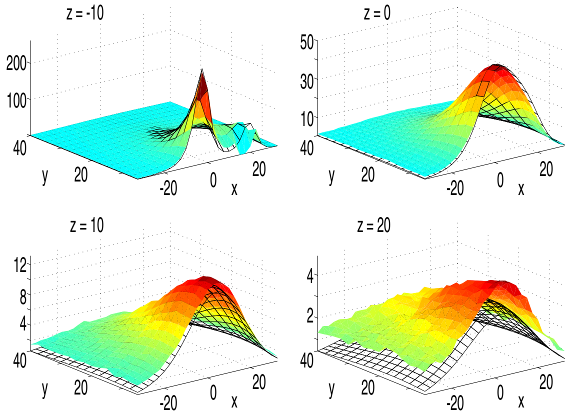


Figure 4.9: Slices from measured (surface) and modified model (mesh) reflections, from Emitter 1 to Receiver 2.

and receiver diodes are distantly spaced, as seen in the top left plot of Figure 4.8 are mainly caused by deviations between the actual emitter characteristic and the model (see Section 3.2).

This misalignment can be eliminated by altering the model so that Emitter 1 when considering the pair Emitter 1 to Receiver 2, is placed at $x = -1.35$cm in stead of $x = 0.65$cm. The result of this is shown in Figure 4.9.

The next section is concerned with a systematic way of adapting the model to the measurements.

4.4 Fitting the Model to Measurements

As described in the previous section, some misfit between the modeled and measured reflections can be found. This section will be concerned with methods to minimize this misfit.

A misalignment of the peak in the reflection intensity was found in Figure 4.8 in the previous section. This misalignment was eliminated by altering the position of the Emitter for that particular emitter/receiver pair. This was the initial step to search for a method capable of doing such a modification of emitter/receiver placement altering.

The method found useful for doing this kind of model adaption is a iterative trial and error method. The parameters of the model that is fitted to the measurements this way are: Emitter position, emitter orientation, receiver position, and receiver orientation.

Initially the emitter and receiver positions and orientations are set to their nominal values in the model (according to actual sensor placement). Then a number of model to measurement fittings are performed, with different variations in emitter position, emitter orientation, receiver position, and receiver orientation. The variation that results in the lowest error in the current iteration is taken as new sensor placement for the model. The other placement variations are discarded, and a new iteration made. The iterations are stopped when there are no significant improvement in the model error.

This model adaption is made for each emitter/receiver pair. The model to measurement fitting is made by least squares, and the model error is found as:

$$e = \sum_{\boldsymbol{c} \in X} (I_{\mathrm{m}}(\boldsymbol{c}, E_{\mathrm{m}}, R_{\mathrm{m}}) - \max(rI(\boldsymbol{c}, E, R), I_{\mathrm{sat}}))^2$$

over r, E_m and R_m, where r is a scalar to compensate for various gains in the physical setup (e.g. ρ_d and receiver amplification), E_m and R_m is emitter and receiver position and orientation in the model respectively, $\boldsymbol{c}$ is the ball position, $I(\boldsymbol{c}, E, R)$ is the measured reflections from the ball at position $\boldsymbol{c}$ and I_{sat} is the level where the receiver saturates.

4.5 Summary

This chapter described the development of a 3D Reflection Map Model, for modeling the intensity of light reflected by a sphere. The model have been evaluated and compared to real measurements of reflected light intensities. The model showed good accordance with the real measurements. Small misalignments was found, caused by deviations between model and real emitter characteristic. A simple method for adapting the 3D Reflection Map Model to the measurements has been proposed, to minimize the effect of the mentioned deviating characteristic.

Chapter 5

3D Position from Reflections

There exists various vision based systems for determining 3D position of objects, for instance [12] and [27]. The vision based systems have very high ratios of acquired to desired information e.g. 1000 or more. The intention with the method presented here is to make a model based position determination system based on an array of fixed positioned infrared LED and photodiodes where the ratio of acquired to desired information is much smaller, e.g. 100 or less. There exists an approach to determine position from reflection using sets of prolate spheroids with emitter and receiver in the focal points [16].

In the present study the theory of light propagation and reflection has been used to make a reflection model for an IR emitter/receiver pair, Chapter 4. This model is used together with a search algorithm to determine the 3D position of a passive object, initially a ball. One application of this 3D position retrieval is a 3D “non-Touch Screen” used to input 3D positions into a computer. For this non-Touch Screen to be handy, pointing using a human hand would be preferable to a ball on a stick.

This chapter describes the development of a method for estimating the position of a ball-shaped object from measurements of the intensity of light reflected by the object itself. The method is tested on modeled light reflections. Results from test with real measurements is given in the next chapter.

5.1 Reflection Space

Prior to trying to use reflections to find the position of the reflecting object, some investigation of the space of reflections is carried out in this section.

As mentioned earlier, it is possible to measure reflections from eight emitters to eight receivers with the hardware available for this project. This gives a total of 64 possible reflection intensity measurements, and the dimensionality of the space of measurements, called the reflection space, might therefore be as high as 64. The 3D Reflection Map Model of Chapter 4 is used for this investigation of the reflection space.

First, the reflection intensities can not be negative. This constraints the space of reflections to non-negative values, and the reflection space, $\mathcal{R}_s$, are hence not a vector space.

$$\mathcal{R}_s = \{\boldsymbol{x} \in \mathbb{R}^n \mid x_i \geq 0\}$$

where n is the dimension of the reflection space and depend on the number of emitters and receivers used.

Second, it is not given that all points with non-negative coordinates can be realized as reflections. The motions of the object from which the light is reflected is three dimensional, and the set of realizable reflections in the reflection space is hence assumed also to be three dimensional.

The question is: How does the set of realizable reflections behave ?

Especially the question about self-cuttings in the realizable reflections, and if there are no self-cuttings how close does the set come to it self where it is “folding”. The self-cutting question is interesting because self-cuttings in the realizable reflections will prevent a “position from reflections” method in giving unique results. The distance between different parts of the set of realizable reflections where the set is “folding” is interesting because it has influence on the continuity of the positions estimated from reflections, when noise is present in the reflections.

Given the positions of the sensors (the emitters and receivers), the set of realizable reflections can be found using the 3D Reflection Map Model as a mapping from $\mathbb{R}^3$ to $\mathcal{R}_s$. This mapping is considered too complex for further work in this direction.

An other approach is to try to do some visualizations of some modeled reflections. Now it is very difficult to deal with visualizations in dimensions higher than three. Therefore the visualizations here are limited to three dimensional plots, with the limitations that gives.

Figure 5.1 and Figure 5.2 shows two examples of how the plane of realized reflections behave for a setup with one emitter and three receivers. The top left sub-plot in each of the figures show the sensor positions, where the circle is the emitter, and the squares are the receivers. The surface plots in the figures are the surface of intensities, where each axis is the logarithm of the output from each of the receivers.

The plotted reflection surfaces are seen to be quite similar for the more distant ones, where $z \geq 16$, where as there are quite some difference for the plots with $z \leq 8$. Tests of further sensor placements shows that there are self-cuttings in most cases. The only ones where no self-crossings have been found are where the three reflections considered

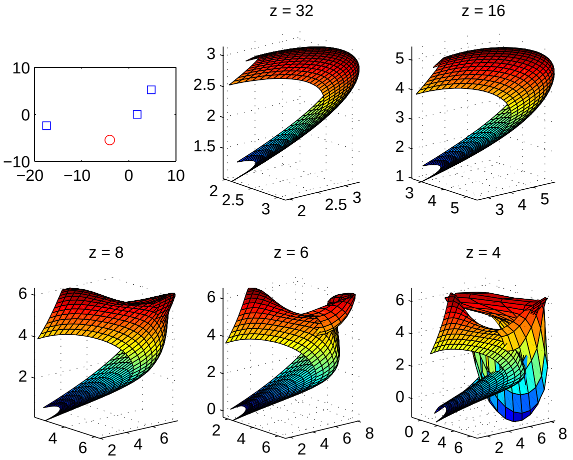


Figure 5.1: Top left: Sensor placement, circle: Emitter, square: Receiver. The surface plots are the surfaces of the logarithms of the reflection intensities. The surfaces is the result of moving the ball-object in a section of a plane over the sensor plane, distance specified by plot titles.

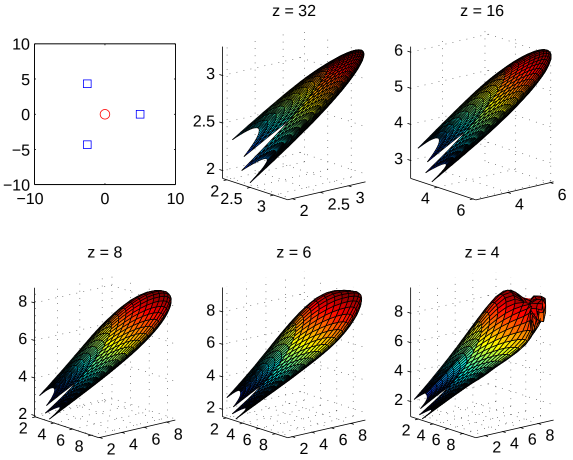


Figure 5.2: Top left: Sensor placement, circle: Emitter, square: Receiver. The surface plots are the surfaces of the logarithms of the reflection intensities. The surfaces is the result of moving the ball-object in a section of a plane over the sensor plane, distance specified by plot titles.

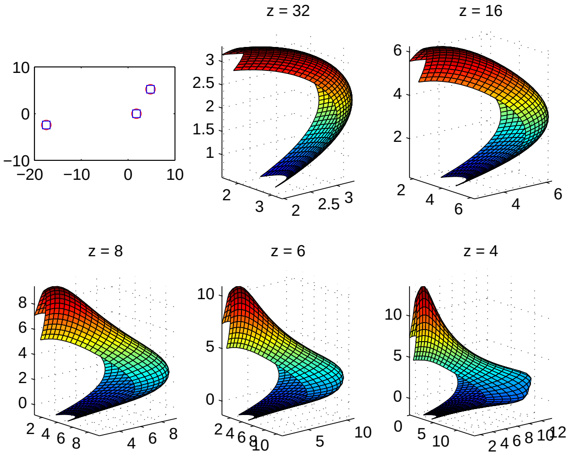


Figure 5.3: Top left: Sensor placement, circle: Emitter, square: Receiver. The surface plots are the surfaces of the logarithms of the reflection intensities. The surfaces is the result of moving the ball-object in a section of a plane over the sensor plane, distance specified by plot titles.

are from three pairs of emitter/receivers where the emitter and receiver in each pair is positioned close together, as shown in Figure 5.3. This does not mean that there are self-cuttings in the realizable set of reflections, but it indicates that for sets of just three reflections the probability of self-crossings are high.

The physics in the reflections and the “room” in high dimensional space e.g. dimension 16 or higher, indicates that it is unlikely that there are self-crossings in the three dimensional set of realizable reflections in this high dimensional space of reflections.

With respect to the distance between “foldings” of the set of realizable reflections, then it is more difficult to give an estimate. Initially the assumption is that this distance is sufficient, to do position estimation from reflections also for measured reflections with noise.

This assumption has shown to be valid from the tests made with position estimation and the “non-Touch Screen” see Chapter 5, as there have not been any results where the position estimation has shown jumps between two pieces of smooth curves. When the positions have been retrievable for the entire test curve, only short jumps out of position and back have been observed, beside the position estimation error.

5.2 Position Retrieval

Based on the 3D Reflection Map Model and measurement of light reflection from a number of emitter/receiver pairs, it is possible to search for the object position that gives reflections from the 3D Reflection Map Model that best fits the measured reflections.

The function chosen to measure goodness of fit is

$$\text{fit}(I_{\text{meas}}) = \|I_{\text{meas}} - I_{\text{model}}\| \tag{5.1}$$

where I_{meas} is a vector of measured reflections and I_{model} is a vector of modeled reflections.

5.2.1 Gradient Methods

The first idea was to apply a simple gradient search method like steepest decent or Newtons method to find the position in the 3D Reflection Map Model which gave the best fit to the measured reflections.

Both methods have been tested and none of them could give satisfactory results. The problem seems to be that the variation in the gradient is such that when the reel position in the search is close to the actual position then the gradient is quite large so that small steps are needed, and when the actual position is far away from the reel position the gradient is almost vanishing. An other problem might raise due to saturation in the receivers, so that the gradient will be wrong when one or more receivers are saturated.

Therefore are other methods not relaying on the gradient searched for.

5.2.2 Naive First Approach

The first non-gradient method tested is a naive search where the measured reflections are compared to the modeled reflections using (5.1).

The principle of the method is most easily described as find the expected reflections in a grid of positions using the 3D Reflection Map Model, then find the one position which fits best, refine the grid in a neighborhood around this position and search again. Stop when either the grid is sufficiently fine or when a predefined maximum number of grid refinements have been done. The principle of the method is visualized in Figure 5.4 in two dimensional space.

The reason for naming this approach “naive” is that lots of calculations are needed in each step. For the two dimensional case in Figure 5.4 m^2 model evaluations are needed in each iteration. When used in a three dimensional space of positions m^3 model evaluations have to be conducted in every iteration making the method very computational

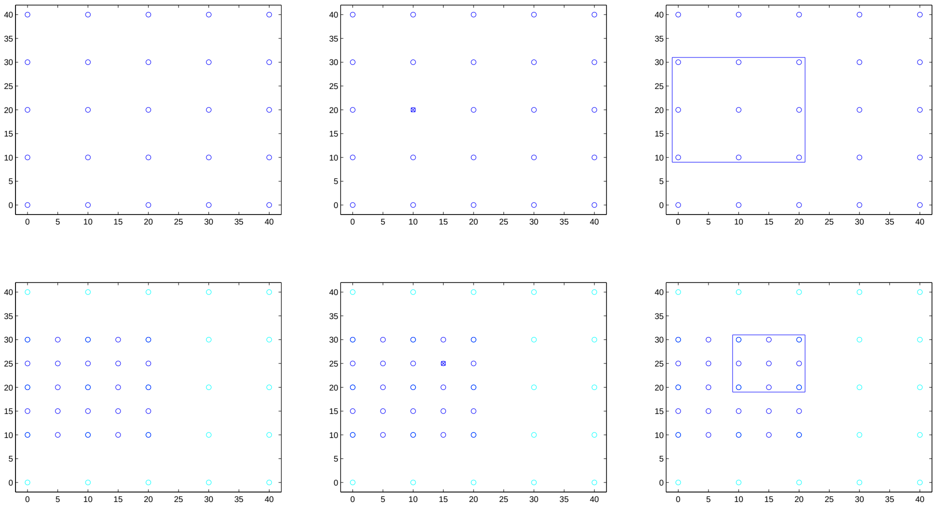


Figure 5.4: Visualization of naive search method, in the top row from left is the initial grid of modeled reflections, in the center is the position of best fit marked, and the right most plot shows the selected neighborhood for grid refinement. The bottom row shows the similar steps on the refined grid in the selected neighborhood.

demanding. m denotes the number of points in the grid in each direction. In Figure 5.4 $m = 5$ which results in 25 model evaluations for a two dimensional problem and 125 model evaluations for a three dimensional problem for each refinement iteration.

The final number of iterations are needed is km where k is the number of iterations performed. One way that this might be reduced is to find an appropriate estimate of the position and not start with the whole region for each search of a new position. This might lower the number of iterations, but still require m^3 model evaluations for each iteration.

5.2.3 Nelder-Mead Simplex Algorithm

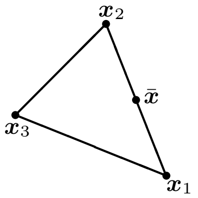
The Nelder-Mead simplex algorithm [23] is an algorithm for finding local minima of a function without using gradient information explicit. The function to be minimized should be of the form

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

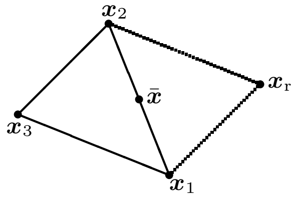
This is exactly the case with the function in (5.1).

The Nelder-Mead simplex algorithm is iterative search algorithm. Each iteration follows the rules:

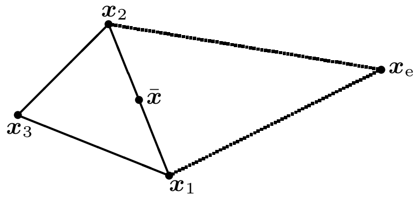
```
0. Setup:
   Set up a simplex of some initial size.

1. Sort:
   Sort the vertices so that
   $f(\boldsymbol{x}_1) \leq f(\boldsymbol{x}_2) \leq \cdots \leq f(\boldsymbol{x}_{n+1})$.
   Find the centroid $\bar{\boldsymbol{x}}$
   of the $n$ best points.
```



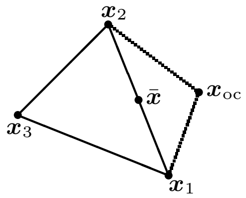
```
2. Reflect:
   Set $\boldsymbol{x}_r = (\mu_r + 1)\bar{\boldsymbol{x}} - \mu_r \boldsymbol{x}_{n+1}$
   If $f(\boldsymbol{x}_1) \leq f_r \leq f(\boldsymbol{x}_n)$ then
           set $\boldsymbol{x}_{n+1} = \boldsymbol{x}_r$ and goto 7.
```



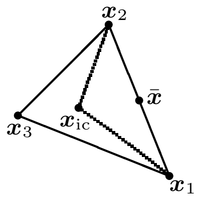
```
3. Expand:
   If $f_r < f(\boldsymbol{x}_1)$
           set $\boldsymbol{x}_e = (\mu_e + 1)\bar{\boldsymbol{x}} - \mu_e \boldsymbol{x}_{n+1}$
   If $f_e < f_r$ then
           set $\boldsymbol{x}_{n+1} = \boldsymbol{x}_e$
   otherwise
           set $\boldsymbol{x}_{n+1} = \boldsymbol{x}_r$ and goto 7.
```



```
4. Outside Contraction:
   If $f(\boldsymbol{x}_n) \leq f_r < f(\boldsymbol{x}_{n+1)}$ then
           set $\boldsymbol{x}_{oc} = (\mu_{oc} + 1)\bar{\boldsymbol{x}} - \mu_{oc} \boldsymbol{x}_{n+1}$
   If $f_c < f_r$ then
           set $\boldsymbol{x}_{n+1} = \boldsymbol{x}_{oc}$ and goto 7.
   otherwise
           goto 6.
```

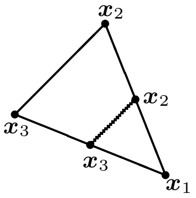


```
5. Inside contraction:
   If $f_r \geq f(\boldsymbol{x}_{n+1})$
           set $\boldsymbol{x}_{ic} = (\mu_{ic} + 1)\bar{\boldsymbol{x}} - \mu_{ic} \boldsymbol{x}_{n+1}$
   If $f_c < f(\boldsymbol{x}_{n+1})$ then
           set $\boldsymbol{x}_{n+1} = \boldsymbol{x}_{ic}$ and goto 7.
```



6. Shrink:

```
For $2 \leq i \leq n+1$
    set $vecx_i = \boldsymbol{x}_1 - (\boldsymbol{x}_i - \boldsymbol{x}_1)/2$
```



7. Stopping condition:

```
If $\max_{i=2,\ldots,n+1} |f(\boldsymbol{x_i}) - f(\boldsymbol{x_1})| < \epsilon_{\mathrm{f}}$
    or $\max_{i=2,\ldots,n+1} \|\boldsymbol{x_i} - \boldsymbol{x_1}\| < \epsilon_{\mathrm{x}}$
    or $N$ function evaluations has been made
then
    exit with $\boldsymbol{x_{\mathbf{min}}} = \boldsymbol{x_1}$
otherwise
    goto 1.
```

Typical parameter values are: $\mu_{\mathrm{r}} = 1$, $\mu_{\mathrm{e}} = 2$, $\mu_{\mathrm{oc}} = \frac{1}{2}$, and $\mu_{\mathrm{ic}} = -\frac{1}{2}$.

When the Nelder-Mead simplex algorithm terminates in step 2 one function evaluation is needed, termination in step 6 requires at most $4+n$ function evaluations in that iteration. The computational time for finding an approximate minimum of the function f can thus be bounded by limiting the number of function evaluations allowed. See e.g. [14] for a more detailed description of the Nelder-Mead Algorithm.

Conserning the convergence properties of the Nelder-Mead Simplex Algorithm then the method is not guaranteed to converge, not even for smooth problems. The problem is stagnation at a nonoptimal point. However, the performance of the method when applied in practice is generally good [14], [20].

5.2.4 Applying the Nelder-Mead Simplex Algorithm to Track an Object

To use Nelder-Mead simplex algorithm for determining the position of the object, setup the simplex in the parameter space for the reflection map model, that is (x, y, z) coordinates for the ball center. In order for the Nelder-Mead Simplex Algorithm to be efficient, a good initial point and an appropriate size for the initial simplex is needed.

In the startup phase tests has shown that choosing the center of the measurable volume as initial point and a simplex size of 1.5 will be appropriate.

The function to be minimized by the Nelder-Mead simplex algorithm is the Euclidean distance between measured and modeled reflections. When the Nelder-Mead simplex algorithm terminates, $\boldsymbol{y} = \boldsymbol{x}_1$ is the estimated object position.

For the second position retrieved start with the first position found, and the same simplex size. For all following positions, the initial position is predicted linearly from the two previous retrieved positions as $\boldsymbol{s}_i = \boldsymbol{y}_{i-1} + (\boldsymbol{y}_{i-1} - \boldsymbol{y}_{i-2})$. The simplex size is set to $\max(0.05, 0.1\|\boldsymbol{y}_{i-1} - \boldsymbol{y}_{i-2}\|)$. Experiments has shown that choosing the initial position and simplex size like this gives a reasonably fast estimation of the new position.

Modeled reflections has been used for a first test of the Nelder-Mead Simplex Algorithm for estimating positions from reflections. Reflections of a ball following the curve:

$$\boldsymbol{c} = \begin{bmatrix} 120\cos(\frac{t}{5}) + 70\cos(\frac{t}{3}) \\ 90\sin(\frac{t}{5}) + 70\sin(\frac{t}{2}) \\ 100\sin(\frac{t}{15}) + 150 \end{bmatrix}, \qquad t \in [0, 30\pi] \tag{5.2}$$

sampled at 189 positions (equidistantly spaced in the parameter t) in 3D space is modeled using the reflection map model of Chapter 4, some noise is added to the modeled reflections, and the Nelder-Mead Simplex Algorithm is used to try to find the positions of the ball from the modeled reflections. The number of function evaluations in this first test is set to a limit of 600. The result of this is shown in Figure 5.5, where the dots are actual positions and the x-marks are estimated positions. The three sub-plots shows the same curve in 3D space from different view-points.

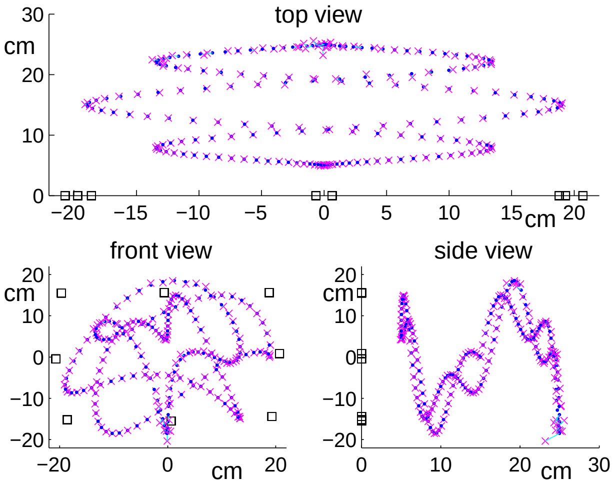


Figure 5.5: Actual ball positions (dots) and retrieved positions (x-marks), pair vise connected by a line. Axis units are cm.

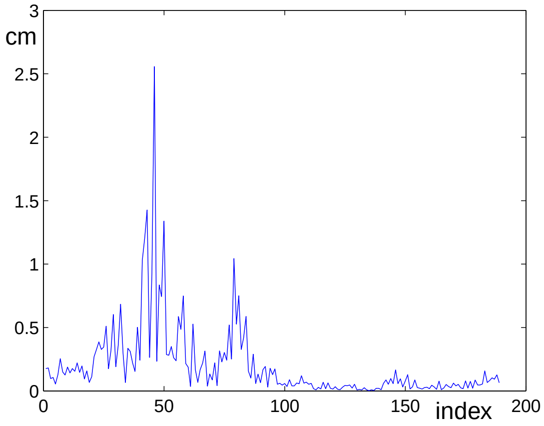


Figure 5.6: The distance between actual ball positions and retrieved positions. x axis is position index, y axis is cm.

As seen from Figure 5.5 there are only small deviations between the actual and retrieved positions, the most obvious deviation is found at the points with largest distance from the plane of the emitter/receiver pairs. Figure 5.6 shows the difference between actual and estimated positions in cm, as a function of position index.

The majority of the deviation between actual and estimated positions in Figure 5.5 and Figure 5.6 originates from the noise added to the modeled measurements. If no noise had been added the deviations would have been much smaller. The reason for adding the noise is to show that the Nelder-Mead Simplex Algorithm has some robustness to noise, and to see if the algorithm would loose track of the object after a period with some deviation. As shown the Nelder-Mead Simples Algorithm can handle noisy data, and the deviations can reduce after a period with larger deviation.

5.3 Summary

Two methods for retrieving the 3D position of a passive object using infrared LED's and photodiodes, based on the 3D Reflection Map Model has been proposed. Both methods searches the parameter space of the 3D Reflection Map Model. One using a naive approach, and the other uses the Nelder-Mead simplex algorithm. The one using the Nelder-Mead simplex algorithm has been tested with modeled light reflection intensities and has shown that the method can successful find positions from reflections.

Chapter 6

Realizing the “non-Touch Screen”

The method of retrieving the 3D position of an object, developed in the preceding chapter is tested by experiment in this chapter. The position of a ball (on a stick) in front of a screen is found based on real measurements of the intensity of light reflected by the ball. It is also shown, that the method is sufficiently robust to allow for exchange of the ball on a stick with a human hand.

The placement of the sensors along the border of the screen is also discussed, in order to find an optimal placement of the sensors (emitter/receiver pairs).

6.1 3D position from Measured Reflections

6.1.1 Physical Setup

The physical setup for laboratory measurements for validating the 3D position retrieval method developed in the previous chapter is sketched in Figure 6.1. The setup consists of eight infrared sensor pairs (SFH485P/BF104F), a positioning device for moving the ball, and a computer with an acquisition interface. The sensors are placed on the boarder of a computer monitor, four in the corners and four halfway down each side. They are all directed towards a point 15cm in front of the screen center. The computer used is a dual Pentium, 700MHz, with NI-DAQ sampling cards (PCI-6071E and 6713).

The 3D Reflection Map Model is calibrated by measuring the reflections from the ball positioned in a 3D grid of $21 \times 18 \times 17$ equidistantly spaced points in front of the sensor

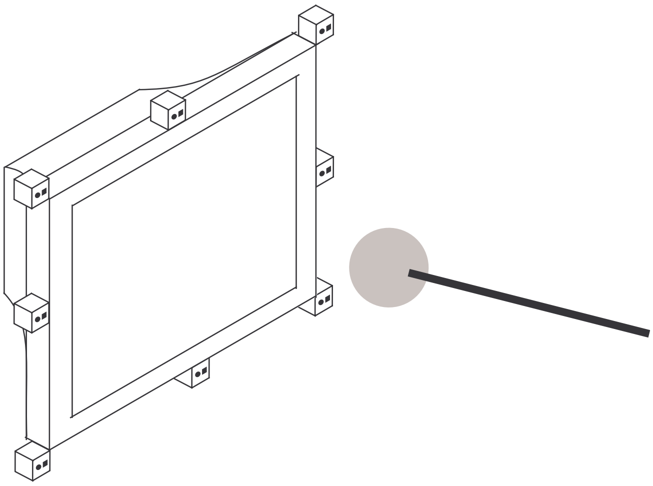


Figure 6.1: Setup for validating the 3D position retrieval method, with eight emitter/receiver pairs around the boarder of the computer screen and the object to be tracked is a ball on a stick.

arrangement, as described in Section 4.4 on page 38. For each position average over 100 reflection measurements are made to reduce noise.

6.2 Robustness and Computation Time

The 3D position retrieval is tested by measuring the reflections from the ball positioned along a curve, and then using the Nelder-Mead simplex algorithm to search for the best fitting 3D position in the Reflection Map Model.

The first test with measured reflections showed that the computation time had to be limited if the position estimation should run in real time on the computer available in the laboratory. There are two obvious means for limiting the computational time: Limit the number of function (model) evaluations and reduce the number of sensor pairs used in the estimation.

First an investigation on the number of function evaluations used by the Nelder-Mead Simplex Algorithm for estimating the positions of the ball along a curve have been made. The result is that limiting the number of model evaluations to about 100 for the first position and 70 for all the following ones gives will only give a small reduction in accuracy.

Figure 6.2 shows the resulting position estimation with the mentioned limits on the num-

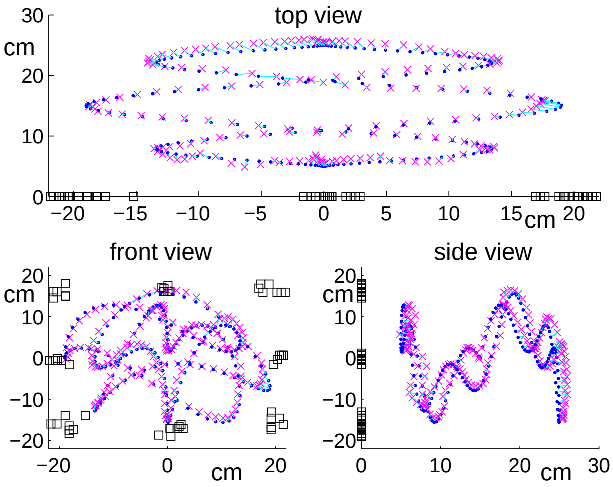


Figure 6.2: Position retrieval using all 64 reflection measurements. Actual ball positions (dots) and retrieved positions (x-marks), pair vise connected by a line.

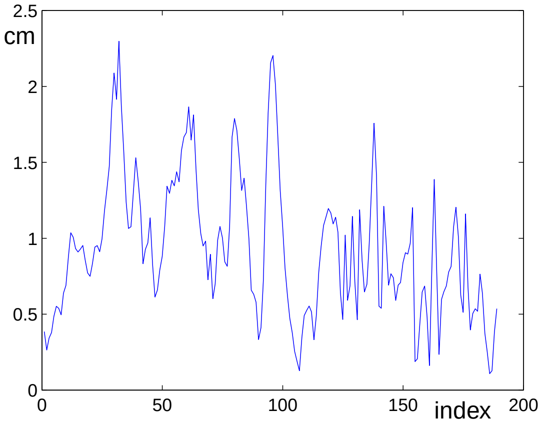


Figure 6.3: The distance between actual ball positions and retrieved positions using all 64 reflection measurements. x axis is position index.

ber of position evaluations, and using reflection measurements from all 64 possible sensor pairs. The dots are actual positions and the x-marks are estimated positions. The three sub-plots shows the same curve in 3D space from different view-points.

Figure 6.3 shows the difference between actual and estimated positions in cm, as a function of position index, and shows that the deviation between actual and estimated positions are within 2.3 cm. The estimation of these 189 positions took 28.4s on the computer available in the laboratory. This gives a mean time of about 150ms to estimate a position from a reflection measurement. The maximal time for estimating one position from 64 measurements is approx. 250ms. This results in a maximum estimation rate of between 4Hz and 6.6Hz if the computer only has to estimate the positions. The computer has to also measure the reflections and do some other tasks like update the screen, this will reduce the frequency with which positions can be estimated.

Investigations on the number of sensor pairs considered has been made, to try to limit the computational time of a position estimation. Figure 6.4 to Figure 6.8 shows the resulting deviations between actual and estimated positions for five different selections of sensor pairs. Figure 6.4 results from selecting all 16 combinations of the four sensors in the corners of the screen. Here the deviation is almost up to 12.5cm, this is six times as high as with all 64 possible measurement. Computation time 6.6s.

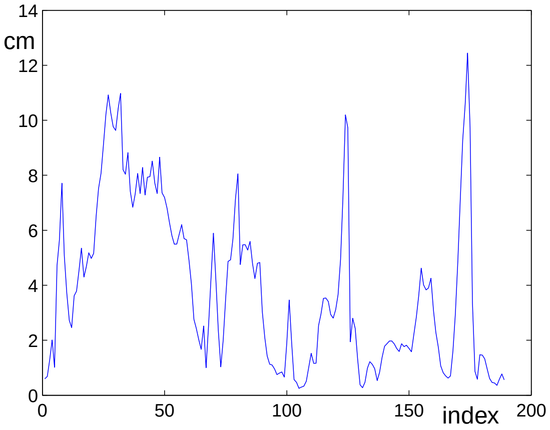


Figure 6.4: The distance between actual ball positions and retrieved positions using the 16 reflection measurements from the four corner sensors. x axis is position index.

Figure 6.5 results from selecting all 16 combinations of the four sensors at the middle of each side of the screen. This is significantly better, though the maximal deviation is as high as 7.4cm there are only a few position estimates that derive more than 4cm from the actual position. Computation time 7.6s.

Figure 6.6 results from selecting only the 8 measurements from the emitter/receiver pairs in the same housing. Now with only eight measurement the deviation is within 3.9cm for all the measurements. Computation time 3.8s.

Figure 6.7 results from selecting the same 8 reflection measurements plus the reflection measurements from neighboring emitter/receiver pairs, 24 combinations in total. This begins to look like the result from using all 64 measurements, the deviations are here within 2.4cm. Computation time 10.9s.

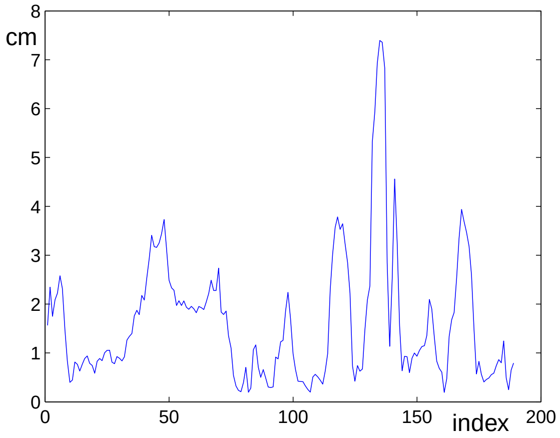


Figure 6.5: The distance between actual ball positions and retrieved positions using the 16 reflection measurements from the four side sensors. x axis is position index.

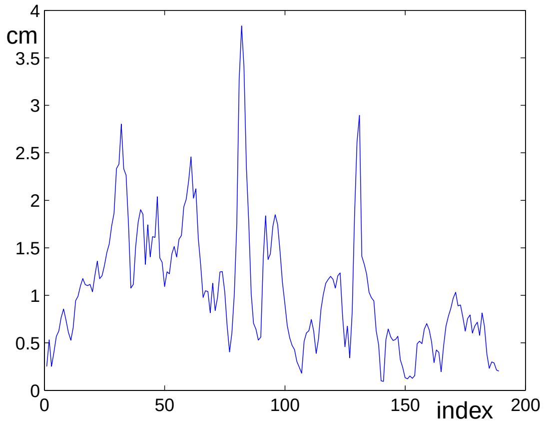


Figure 6.6: The distance between actual ball positions and retrieved positions using the 8 reflection measurements from the eight individual sensors. x axis is position index.

Figure 6.8 results from selecting further the remaining combinations from the four sensors on the middle of each screen side except the two from middle of left side to middle of right side of the screen, 34 measurements in total. The deviation between actual and estimated positions are here within 2.2cm, which is actually better than the result obtained using all 64 possible measurements. Computation time 16.3s.

The reason that the result in Figure 6.8 using only 34 reflection measurements, is slightly better than the one in Figure 6.3 using all 64 possible measurements, is that the mea-

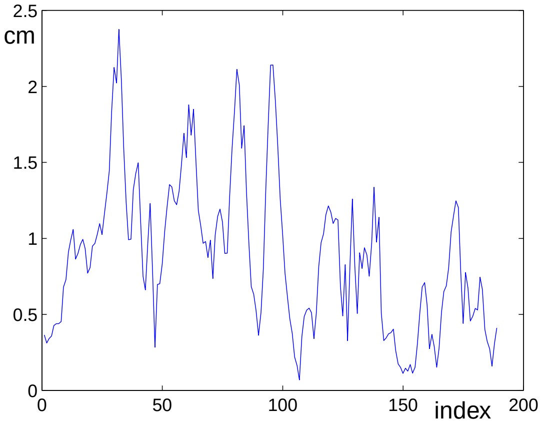


Figure 6.7: The distance between actual ball positions and retrieved positions using 24 reflection measurements from neighboring sensors. x axis is position index.

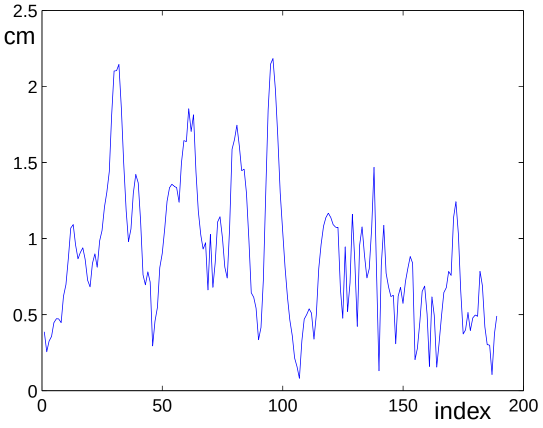


Figure 6.8: The distance between actual ball positions and retrieved positions using 34 selected reflection measurements. x axis is position index.

sured reflections are quite small for distantly placed emitter/receiver pairs, so that measurement noise and model deviation exceeds a level where reliably position estimation can be made. The same effect can be seen when comparing Figure 6.4 and Figure 6.5, each using 16 measurements. The sensor placements are in both cases the corners of a rectangle, but the distance between emitters and receivers are different.

The setup with 34 reflection measurements used for estimating the position is selected for a test implementation of the non-Touch Screen. Figure 6.9 shows top, front and side

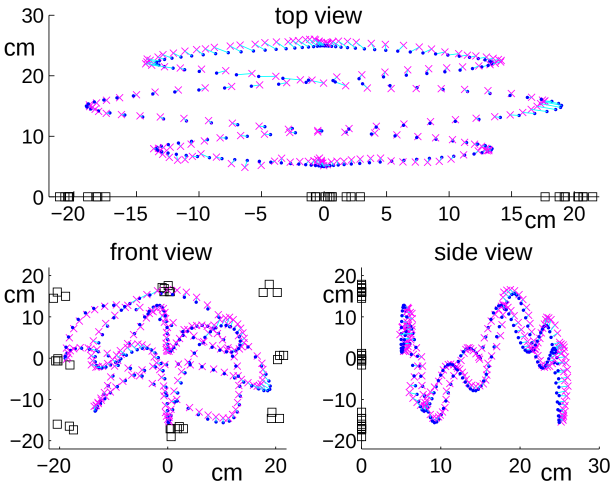


Figure 6.9: Position retrieval using the 34 selected reflection measurements. Actual ball positions (dots) and retrieved positions (x-marks), pair vise connected by a line.

view of both retrieved positions (x-marks) and actual ball position (dots), the actual ball position are connected to the corresponding estimated position to show the alignment. The axis units on the figure are centimeters.

Figure 6.10 shows the position deviation between actual ball positions and retrieved positions for the same curve if the ball had been moving twice as fast along the curve (every 2'nd point used). Still with distance between actual and retrieved position below 2.2cm.

6.3 The Non-Touch Screen

It is possible to realize a “3D non-Touch Screen”, using the same sensor setup of the previous section, and use a human hand as pointing device. The precision of the 3D position detected degrades slightly due to disturbance reflections from the human arm, and only averaging over 38 measurements. With the hardware used the 3D positions can be obtained with a rate of $5Hz$.

Figure 6.11 and Figure 6.12 shows photos of the “non-Touch Screen” in action. The courser on the computer screen is a disc, that changes size depending on the depth coordinate i.e. when the hand is close to the screen a big disc is shown and when the hand is further away from the screen a smaller disc is shown.

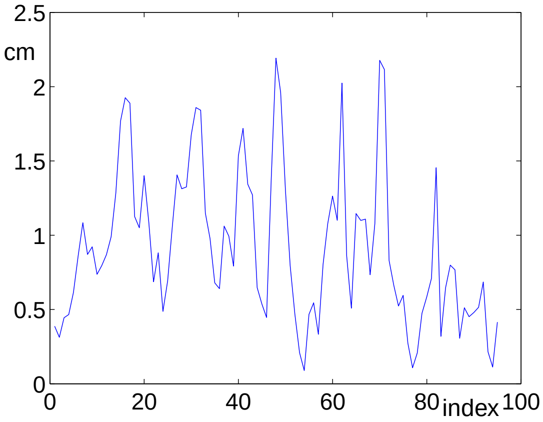


Figure 6.10: The distance between actual ball positions and retrieved positions using the 34 selected reflection measurements, for a faster object motion (every 2’nd point used).

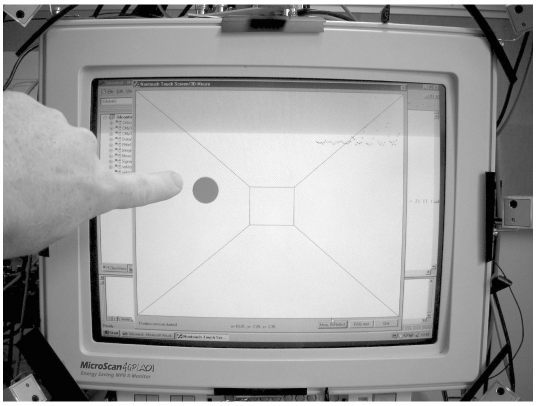


Figure 6.11: The “non-Touch Screen” with the hand close to the screen.

6.4 Optimal Sensor Placement

The results of the robustness investigation of Section 6.2 showed that the precision with which the positions could be retrieved, to some extend depended on which measurements were used in the position retrieval algorithm. Especially there were large differences when choosing the four sensor pairs in the corners and the four sensor pairs on the sides, despite the structural similarities in the two setups.

In this section an investigation on where to place the sensors optimally along the boundary of the screen is conducted.

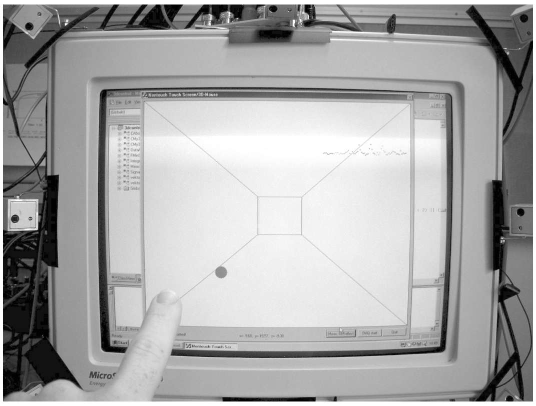


Figure 6.12: The “non-Touch Screen” with the hand distant from the screen.

6.4.1 Prolate Spheroids

In [16] it is suggested that the optimal position of sensors can be found by use of prolate spheroids, one for each combination of emitter and receiver. The spheroids are constructed by rotating an ellipse with the emitter in one focal point end the receiver in the other focal point, around the semi-major axis.

The assumption is that the contour surfaces of the reflected light intensities are prolate spheroids with an emitter and a receiver in each focal point. Using the 3D Reflection Map Model from Chapter 4 and measured data the validity of this assumption can be tested. Figure 6.13 and Figure 6.14 shows the contour surfaces for modeled respectively measured reflection intensities for six different levels, 5, 10, 20, 40, 60, and 80.

From Figure 6.13 and Figure 6.14 it is obvious that the assumption that the contour surfaces are prolate spheroids with emitter and receiver in the focal points are not quite good.

Therefore an other method for determining optimal placement the of sensors are searched for.

6.4.2 Search the Model for Optimal Sensor Placement

Another way to try to find an optimal placement of sensors is to use the 3D Reflection Map Model of Chapter 4 to try to find an optimal placement of the sensors, by searching trough possible sensor placements.

The first search for an optimal sensor position have been made by modeling the reflec-

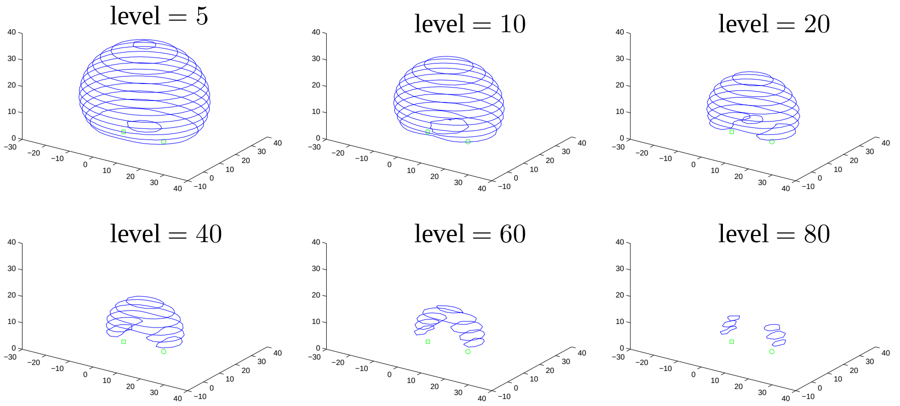


Figure 6.13: Contour surfaces for a pair of emitter and receiver from modeled reflection intensities, at six different levels.

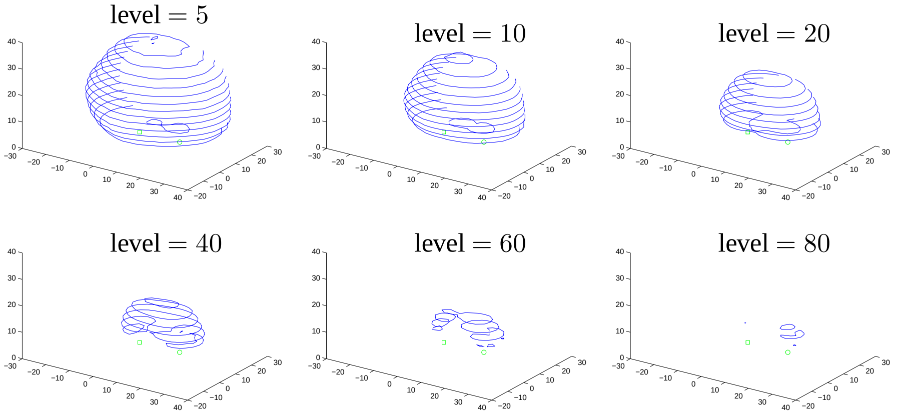


Figure 6.14: Contour surfaces for a pair of emitter and receiver from measured reflection intensities, at six different levels.

tions from a ball following the curve used to test the position estimation method (5.2) in Chapter 5, add some noise to the modeled reflections, and then use the Nelder-Mead Simples Search Algorithm as described in Section 5.2 to estimate the positions from the “noisy” reflections. Once the positions have been estimated from the reflections for a number of possible sensor setups, the deviation in positions between actual and estimated positions is found as

$$d_i = \|C_{\text{actual}\,i} - C_{\text{estim}\,i}\|$$

where i index the positions.

Based on these distances a measure of goodness is used to rank the individual sensor placements. Three different measures of goodness are:

$$D_1 = \sqrt{\sum d_i^2} \tag{6.1}$$

$$D_2 = \max_i(d_i) \tag{6.2}$$

$$D_3 = \text{mean}(d_i) \tag{6.3}$$

Where D_1 measures the Euclidean norm of the position deviation vector, D_2 measures the maximum deviation and D_3 measures the mean value of the deviations. For all three measures of goodness a small value indicate better performance than larger values, but it is not possible to compare values from the different measures.

The measure D_2 should be chosen if the largest error on the position estimation has to be kept as small as possible. D_3 should be chosen if only the mean value of deviations is of interest, and no attention is paid to a few large deviations. The measure D_1 could be taken as a compromise of D_2 and D_3, favoring a small mean and simultaneously punishing large deviations.

With D_1 as measure of goodness the optimal sensor placement found is shown in Figure 6.15 for four, six, and eight sensor pairs respectively. Notice, that this result is obtained with only one (arbitrarily chosen) test curve of positions.

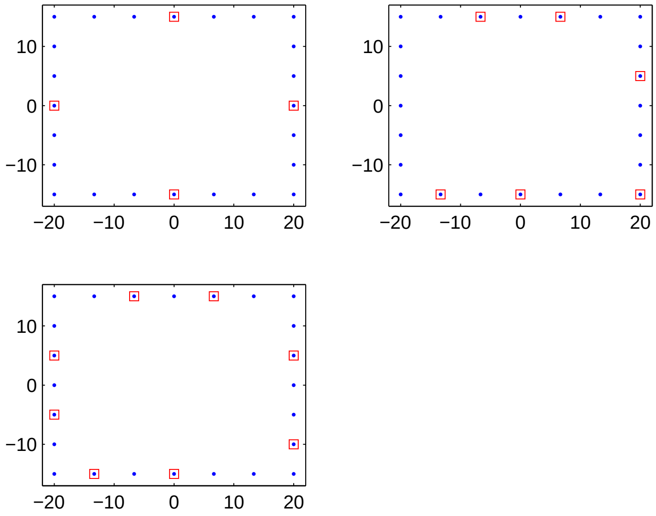


Figure 6.15: Optimal sensor placement of sensors found by evaluating the model along the curve (5.2), with four sensor pairs (top left), six sensor pairs (top right), and eight sensor pairs (bottom left).

This search method is quite time consuming, since it is necessary first to model the reflections for each possible sensor setup, and then run the position estimation also for each sensor setup.

6.4.3 Algebraic Criteria for Optimal Sensor Placement

The 3D Reflection Map Model of Chapter 4, can be used when trying to find an algebraic criteria for optimal sensor placement. This model search requires a measure for goodness of a particular sensor placement. Searching trough all possible sensor placements and evaluate the measure of goodness will yield a solution.

A first idea of a measure of goodness was to maximize the smallest gradient of reflections amongst different sensor placements. But the norm of $\nabla I(x, p)$ where x is object position and p is sensor placement will reduce in size with increased distance from sensors to object. Simply because the reflected intensities decrease with increasing distance. This will result in placing all sensors near the middle position, because this will give the smallest distance to the object position furthest away from the individual sensor pair.

A better measure for goodness of sensor placement is:

$$P_{\text{opt}} = \arg \max_{p \in \mathcal{P}} \min_{x \in X} \underline{\sigma}(\nabla I(x, p)) \tag{6.4}$$

where $\underline{\sigma}$ denotes the smallest singular value. Here the difference in local distortion of the space is measured. A small value indicates that the space is becoming “flat” in the sense that the set of gradients in a single point has only small variation in one direction compared to (at least one) other directions. Also this measure is fairly independent of the distance between the object and the sensors.

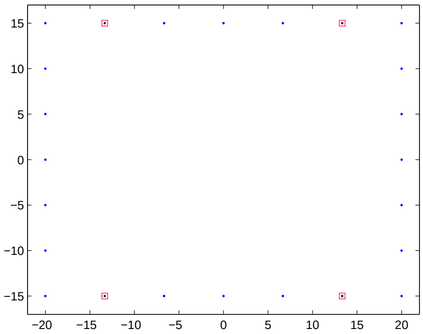


Figure 6.16: Optimal sensor placement for four sensors (squares) amongst the possible sensor placements (dots), with respect to optimality condition (6.4).

For four sensors this results in a sensor placement as shown in Figure 6.16, where the dots are possible sensor positions (for placing the sensors along the border of a computer screen) and the squares are the optimal placement for the sensors. The sensor placement in Figure 6.16 is found with sensors having their normal perpendicular to the plane where the sensors are placed. The sensors are place near the corners as expected. This sensor placement was found as particularly bad in [16], as mentioned in the previous subsection.

With six or eight sensors the optimal placements are shown in Figure 6.17. Again the sensor normals are orthogonal to the plane where the sensors are placed.

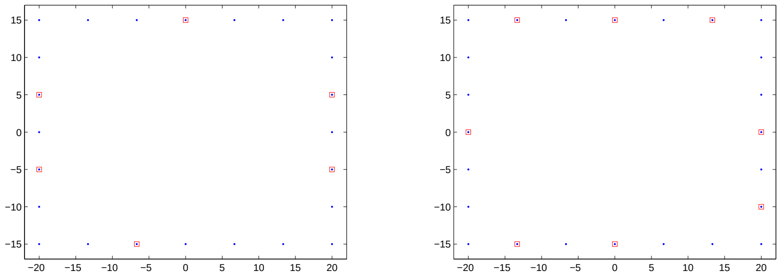


Figure 6.17: Optimal sensor placement (squares) amongst the possible sensor placements (dots), left: Six sensors, right: Eight sensors, both with respect to optimality condition (6.4).

Since both emitters and receivers are most powerful respectively sensitive in the direction near their normal direction, a new search for optimal placement of the sensors have been made with the sensors angled 45° towards the center of the area of interest for position determination, as shown in the top left plot of Figure 6.18. The reason for this is to try to maximize the amount of optical power emitted into the area of interest for position retrieval.

With the angled sensors the optimal positioning of sensors with respect to (6.4) is shown in Figure 6.18.

Comparing Figure 6.16 and Figure 6.18 shows that the optimal sensor position for four sensors do not change when changing the direction of the sensor normals. Both for six and eight sensors the positioning of sensors change, as seen when comparing Figure 6.17 and Figure 6.18.

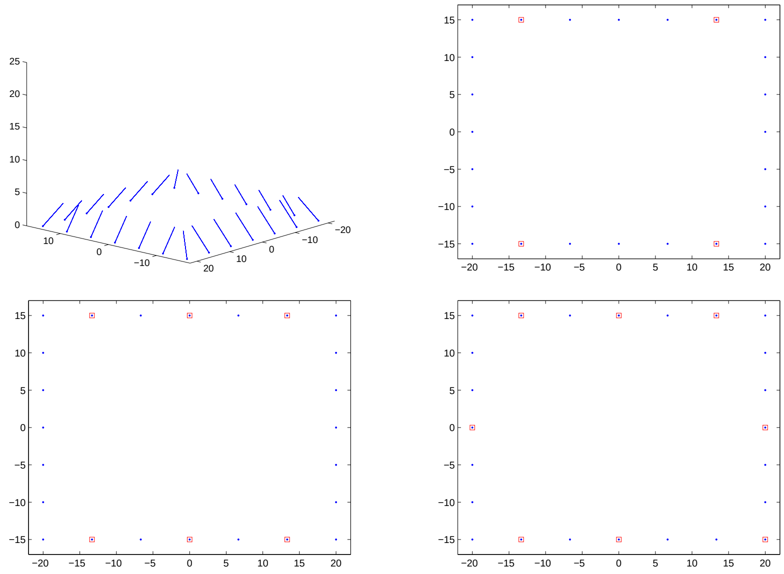


Figure 6.18: Top left: Sensor normals for optimal placement search. Optimal sensor placement (squares) amongst the possible sensor placements (dots). Top right: Four sensors, bottom left: Six sensors, bottom right: Eight sensors. All with angled sensors and with respect to optimality condition (6.4).

6.5 Validation of Optimal Sensor Placements

The experience from tests of position retrieval gave some a somewhat different idea of what the optimal sensor placement, than those found in the previous section. In this section the curve used to test the position retrieval procedure in Chapter 5 is used to test the different optimal sensor placements found in the previous section, the placement used in Chapter 5, and some other sensor placements that arise along the tests.

To differentiate the individual sensor placements the errors needs some sort of quantification of the over–all deviation. Depending on what criteria is considered to be most important the goodness of the sensor placements can be measured using either (6.1), (6.2) or (6.3).

With four sensors the sensor placements tested is shown in Figure 6.19.

The setup 4_1 is the output of the search for optimal sensor placement from the previous section. Setup 4_2 and setup 4_3 are chosen because they can be realized as a subset of sensors from the position retrieval measurement tests from Chapter 5. Setup 4_4 is a test to see how slight translation of sensor positions effect the position retrieval.

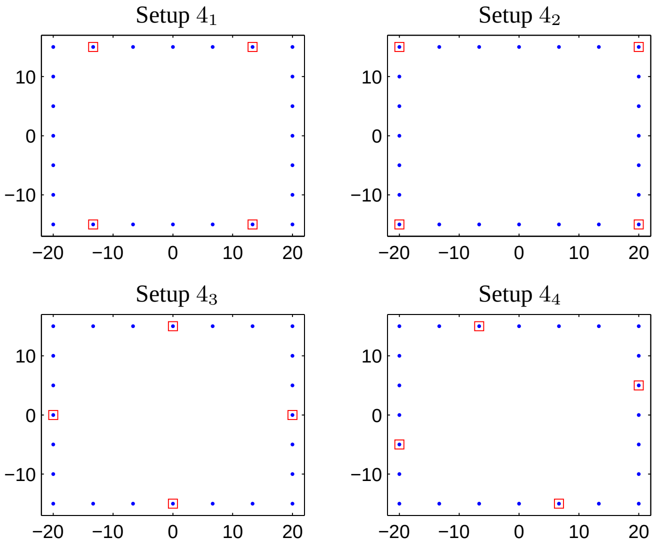


Figure 6.19: Four different sensor placements with four sensors for validation of optimality of sensor placement.

The four sensor placements in Figure 6.19 are tested by modeling the reflections from a ball following the same curve as used in the position retrieval test in Chapter 5 and then, after adding some artificial noise to the modeled reflections, trying to retrieve the positions of the ball by the method proposed in Chapter 5. This have been done both with sensor normal vectors being perpendicular to the plane containing the sensors, and for sensor normal vectors angled as previously described (sketched in Figure 6.18).

The deviation distance between the actual and retrieved positions are shown in Figure 6.20 for sensors with normals perpendicular to the plane containing the sensors.

The deviation distance between actual and retrieved positions, for sensors placed such that their normals are perpendicular to the plane of sensors, in Figure 6.20 shows that for sensor setup 4_2 and setup 4_4 are particularly bad. For both these the position retrieval breaks down, and the estimated positions gets to be completely wrong. The sensor setup 4_1, found to be the best according to the criteria (6.4) in the previous section, is found to perform significantly worse than the sensor setup 4_3.

For sensors with angled normal vectors (as sketched in Figure 6.18) the deviation distance is shown in Figure 6.21.

When placing the sensors such that they have normal vectors angled into the field of interest for the position retrieval, the sensor setup 4_2 and setup 4_4 now performs significantly better than for sensors with normals perpendicular to the sensor plane, as shown

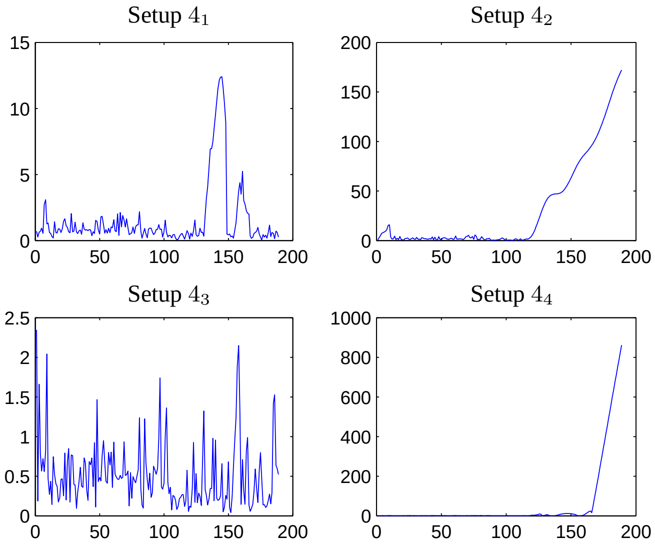


Figure 6.20: Distance between actual positions and retrieved positions for the four sensor placements, with sensor normals perpendicular to the plane of sensors.

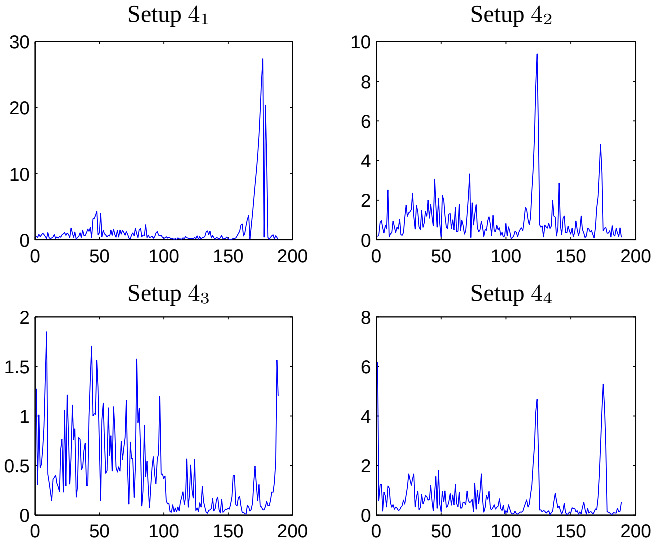


Figure 6.21: Distance between actual positions and retrieved positions for the four sensor placements, with angled sensors.

in Figure 6.21. Now they actually performs better than sensor setup 4_1 which was found to be optimal in the previous section. Again sensor setup 4_3 gives the least deviation between actual and retrieved positions.

Comparison of Figure 6.20 and Figure 6.21 shows that angleing the sensors for sensor setup 4_1 results in larger deviations on the retrieved positions for the test curve. For the other three setups 4_2, 4_3, and 4_4 the angeling of sensors results in smaller deviation distances, most significantly for setup 4_2 and 4_4 which now performs better than setup 4_1 did for sensors with normals orthogonal to the plane of sensors.

Table 6.1 shows the result of quantifying the deviations in Figure 6.20 and Figure 6.21, by the measures of goodness (6.1) (6.2) (6.3).

Sensor placement	D_1	D_2	D_3
Setup 4_1	40.50	12.4	1.583
Setup 4_2	770.9	172.1	31.26
Setup 4_3	9.039	2.345	0.5141
Setup 4_4	2480.	861.6	57.44
Setup 4_1 angled	55.62	27.40	1.547
Setup 4_2 angled	21.32	9.385	1.001
Setup 4_3 angled	8.066	1.849	0.429
Setup 4_4 angled	15.61	6.186	0.6528

Table 6.1: Quantification of error in distance between actual and estimated positions for the eight sensor setups with four sensors.

The best sensor placement is, according to all three measures of goodness (6.1), (6.2) and (6.3) listed in Table 6.1, sensor setup 4_3 where the sensors are placed on the middle of each of the four sides, and the one with angled sensors performs slightly better than the one with out angled sensors. It is also worth noting, that the two sensor placements found to be optimal in the previous section, setup 4_1 without and with angled sensors, for all measures of goodness listed in Table 6.1 gives the poorest results amongst the six setups that does not completely loose track of the object.

With six sensor pairs the evaluated sensor placements are the ones shown in Figure 6.22.

Setup 6_1 and setup 6_2 are the results from the previous section of searching for optimal sensor placements for six sensors without angled sensors and with angled sensors, respectively. Setup 6_3 and setup 6_4 are setups similar to the best setup for four sensors (setup 4_3, see Figure 6.19), and setup 6_4 is also almost identical to setup 6_1.

The deviation distance between the actual and retrieved positions for six sensor setups are shown in Figure 6.23 for sensors with normals perpendicular to the plane containing the sensors, and in Figure 6.24 for sensors with angled normal vectors. Table 6.2 shows the result of quantifying the deviations in Figure 6.23 and Figure 6.24, by the measures

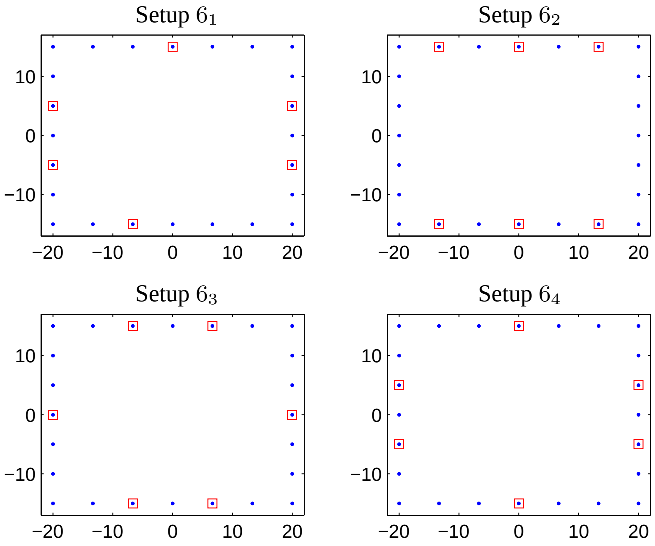


Figure 6.22: Four different sensor placements with six sensors for validation of optimality of sensor placement.

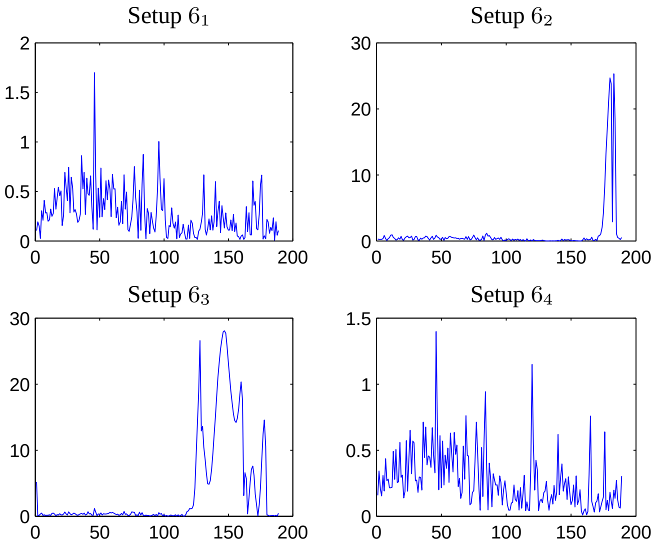


Figure 6.23: Distance between actual positions and retrieved positions for four sensor placements, with sensor normals perpendicular to the plane of sensors.

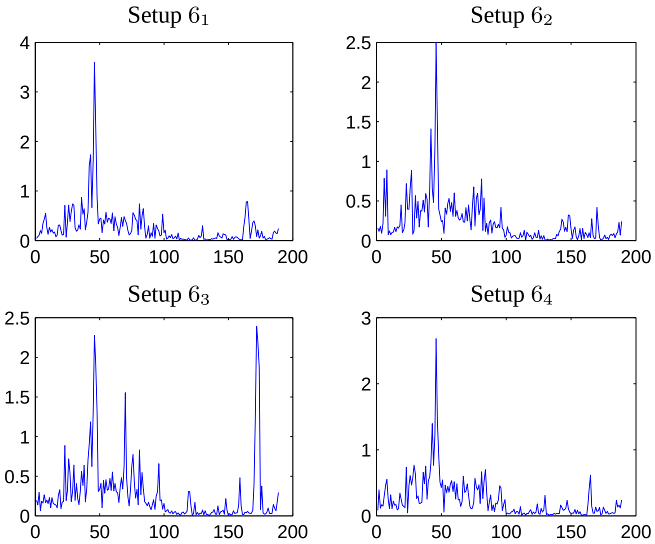


Figure 6.24: Distance between actual positions and retrieved positions for four sensor placements, with angled sensors.

Sensor placement	D_1	D_2	D_3
Setup 6_1	5.0625	1.698	0.2844
Setup 6_2	73.054	33.67	1.903
Setup 6_3	113.81	28.08	4.066
Setup 6_4	4.8001	1.398	0.2759
Setup 6_1 angled	6.566	3.595	0.2701
Setup 6_2 angled	6.909	2.562	0.3451
Setup 6_3 angled	6.760	2.391	0.2809
Setup 6_4 angled	5.403	2.683	0.2455

Table 6.2: Quantification of error in distance between actual and estimated positions for the eight sensor setups with six sensors.

of goodness (6.1) (6.2) (6.3).

Comparing Figure 6.23 and Figure 6.24 it is first noted that the track of the ball is never completely lost, though their are some considerably deviations for the setups 6_2 and 6_3 without angled sensors. The four sensor placements with angled sensors performs quite similarly, as can also be seen from Table 6.2 with a small favor to setup 6_4 when considering measures D_1 or D_3, for measure D_2 the favor goes to setup 6_3. The setups 6_1 and 6_4 without angled sensors both performs better than all setups with angled sensors when measuring both D_1 and D_2. It is also worth noting, that setup 6_2 without angles

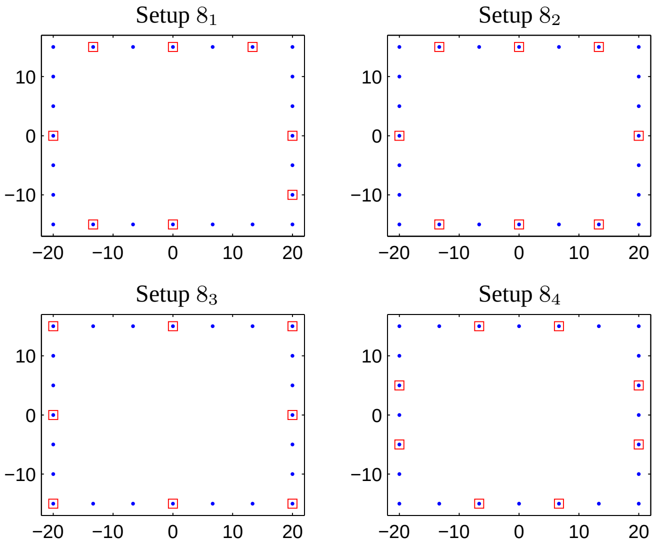


Figure 6.25: Four different sensor placements with eight sensors for validation of optimality of sensor placement.

sensors gives the smallest value for the measure D_3, even though this setup have the largest maximal deviation (measure D_2). Surprisingly the setups with angled sensors are outperformed on all measures of goodness.

With eight sensor pairs the evaluated sensor placements are the ones shown in Figure 6.25.

Setup 8_1 is the sensor placement found to be optimal in Section 6.4.2. Setup 8_2 is a modification of setup 8_1 to give a symmetric sensor placement and to be close to setup 6_2 (the optimal one from searching for sensor placements for six angled sensors). Setup 8_3 is the setup used in Chapter 5 for realizing the non-Touch Screen, and setup 8_4 is found by trying to come close to the best performing four sensor setup 4_3.

The deviation distance between the actual and retrieved positions for eight sensor setups are shown in Figure 6.26 for sensors with normals perpendicular to the plane containing the sensors, and in Figure 6.27 for sensors with angled normal vectors. Table 6.3 shows the result of quantifying the deviations in Figure 6.26 and Figure 6.27, by the measures of goodness (6.1) (6.2) (6.3).

Comparing Figure 6.26 and Figure 6.27 it is first noted that they generally perform quite equal, and also quite good compared to the previously tested setups with four and six sensors. From Table 6.3 setup 8_3, with both angled and non angled sensors, are found to perform slightly worse than the other setups for all measures of goodness. The best

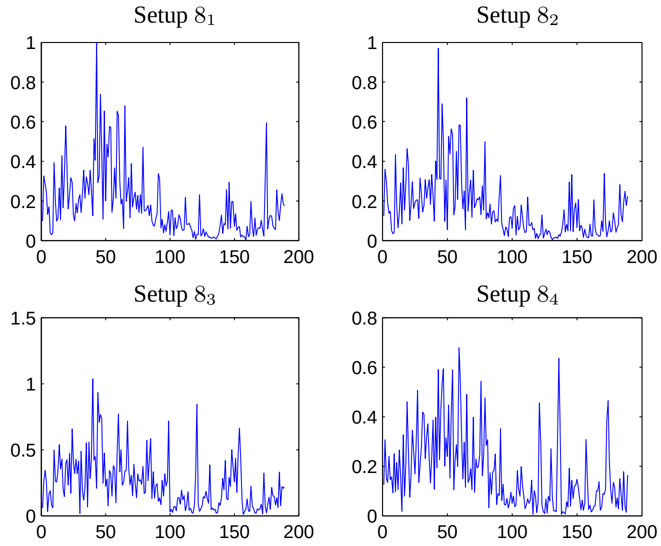


Figure 6.26: Distance between actual positions and retrieved positions for four sensor placements, with sensor normals perpendicular to the plane of sensors.

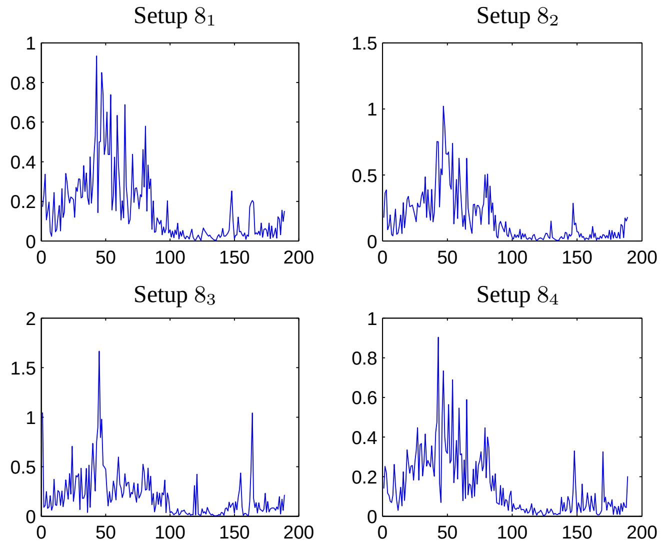


Figure 6.27: Distance between actual positions and retrieved positions for four sensor placements, with angled sensors.

Sensor placement	D_1	D_2	D_3
Setup 8_1	3.308	0.9979	0.1759
Setup 8_2	3.099	0.970	0.1648
Setup 8_3	4.303	1.037	0.2428
Setup 8_4	3.167	0.6796	0.1750
Setup 8_1 angled	3.278	0.9336	0.1621
Setup 8_2 angled	3.458	1.019	0.1664
Setup 8_3 angled	4.234	1.666	0.2068
Setup 8_4 angled	2.953	0.9037	0.1498

Table 6.3: Quantification of error in distance between actual and estimated positions for the eight sensor setups with eight sensors.

setup is found to be setup 8_4 with angled sensors when considering measures D_1 and D_3 and for measure D_2 the same setup, but without angled sensors.

6.6 Summary

In this chapter it has been shown that the method for retrieving positions from reflections developed in Chapter 5 can be used with measured reflection intensities. For the precision of the individual position estimates to be as low as possible it has been shown that it is not always advantageous to use as many measurements as possible. When the measured intensity of reflected light are very low, noise and model deviations tends to blur the position estimation.

The method for retrieving positions from reflections has good robustness against objects not being ball-shaped, as shown in Section 6.3. There the “ball on stick” object was successfully exchanged with a human hand as object to be traced.

Also some investigation on optimal sensor placement has been made. First it was shown that the level surfaces for the intensity of reflected light from a ball does not form prolate spheroids, as assumed in [16] for a proposed method of finding optimal sensor placement. This was shown for both measured reflections and modeled reflections using the 3D Reflection Map Model. After that two search methods was proposed for searching for optimal sensor placements using the 3D Reflection Map Model of Chapter 4. One search method models reflections and try to retrieve the positions from the modeled reflections, and the other evaluates an algebraic criteria for goodness of sensor placement. Both methods searches trough all possible sensor placements. Validation of the candidates for optimal sensor placements have been compared using a modeled curve of positions. Further work is needed before a final conclusion can be drawn with respect to what placement of sensors is optimal.

Chapter 7

2D Position from Reflections

During my Ph.D. I paid a visit to the Department of Electronics at Alcalá University in Spain. There it was proposed to me, that the light sensors which i worked on could be used for an input device for an electrical wheelchair. The input device should measure the 2D position of a human head with the purpose of controlling the movements of the wheelchair.

This chapter is concerned with the methods for estimating the position of an object in 2-dimensional space. The method developed in Chapter 5 together with the 3D Reflection Map Model of Chapter 4 could be used for this purpose, as shown in Section 7.3, but has some disadvantages. A different method for estimating the 2D position of the object is proposed in Section 7.4. This proposed method does not have the same disadvantages as the method of Chapter 5.

For an application of the 2D position retrieval method developed here, see Chapter 8 where a head sensor is implemented with the purpose of controlling the movements of a wheelchair by head movements.

With the application of detecting the position of a human head in mind (limited range of movements), a general method for detecting 2D position of an object is developed.

7.1 Modeling Object and Object Movements

A sphere has been chosen as the object for the two-dimensional position estimation procedure. With this choice the 3D Reflection Map Model of Chapter 4 can be used to model the reflections.

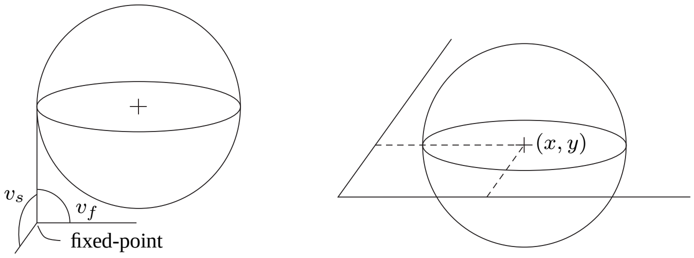


Figure 7.1: Left: First movement model with movements on a sphere-surface, the position of the object is given by the two angles. Right: Simpler movement model, where the object is assumed to move on a planar surface.

The movements of the sphere object is limited to a two-dimensional surface. The surface is assumed known in advance. Two models of the object movements has been tested. The first approach was to try to make a model where the sphere moved around a point in a fixed distance, resulting in a surface like part of a sphere. The position of the object was described by two angles: A forward–backward angle and a sideways angle. This is shown in the left hand side of Figure 7.1. This model has the disadvantage that the same position of the sphere can be obtained from different angles, so that the angles can not be determined uniquely from the reflections.

A simpler model of the object movements is to assume that the sphere center moves on a plane, as shown in the right hand side of Figure 7.1. This gives the advantage that the object position (soordinate in the plane) is the direct input to the 3D Reflection Map Model, and also the direct output of a position retrieval procedure like the one described in Chapter 5.

7.2 Physical Sensor Setup

Different sensor configurations has been tested. The first tests was carried out with eight emitter/receiver pairs on a row, equally spaced with 5cm from sensor to sensor. This setup was functionally, but showed potential for improvements.

The second setup tested includes two rows of four infrared emitter/receiver pairs mounted on one side of the object. Figure 7.2 shows the position of the infrared emitter/receiver pairs (squares) and the sphere, the units is centimeters. The intention with placing the sensors on an arc rather than on a line is to obtain approximately the same distance from center of the object to each sensor, when the object is in the center of the area of interest for position estimation.

This setup can be further simplified by only using one row of four sensors, and place the

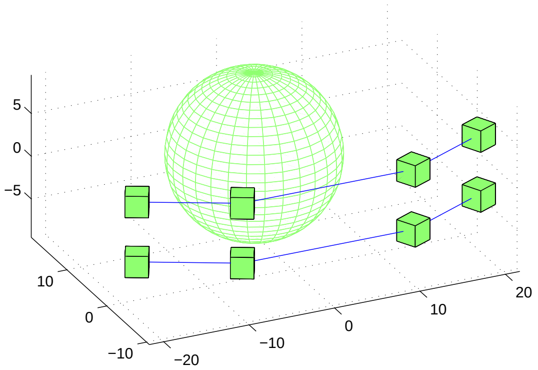


Figure 7.2: The physical placement of the two rows of four emitter/receiver pairs (squares) relative to the sphere. Units are cm.

row of sensors in the plane of the sphere center movement, as shown in Figure 7.3.

With the data acquisition system, each receiver is capable of measuring the reflected light intensity of light emitted from each emitter. Because the object in certain positions will prevent light from the emitters on the left side of the object to be visible at the receivers on the right side of the object no cross measurements are made from left to right side or vise versa. This means that only eight intensities are measured, and the position determination are based on these eight measurements. Naming the emitter/receiver pairs 1–4 from left to right, as done in Figure 7.3. The measured reflections are in this order: E1–R1, E2–R1, E1–R2, E2–R2, E3–R3, E4–R3, E3–R4, E4–R4.

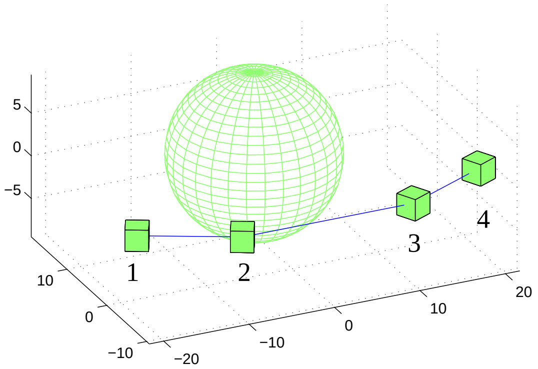


Figure 7.3: The physical placement of one row of four emitter/receiver pairs (squares) relative to the sphere. Units are cm.

The setup with one row of sensors shown in Figure 7.3 is selected for further development, as the initial results with this setup is very similar to the results obtained with two rows of sensors shown in Figure 7.2. Both these setups perform better than the setup used for the initial tests with eight sensors on a line.

7.3 2D Position Retrieval Method 1

The first method proposed for finding the position of an object is based on the 3D Reflection Map Model of Chapter 4 and the position retrieval method based on the Nelder-Mead Simplex algorithm form Chapter 5. Using this method requires a model of the object movements.

Calibration is necessary to fit measurements to the 3D Reflection Map Model. This have been done using a styrene ball of radius 8cm (best object available). The calibration is not as precise as it could be with the positioning device, which was not available at the time. So a manual positioning of the object has been used for the calibration.

Figure 7.4 shows the actual positions (dots) and the retrieved positions after calibration (x-marks), lines connects corresponding positions added for tracking purpose only. The measurements used for calibration are averaged over 800 samples.

The scales obtained in the calibration is divided by two (value found by laboratory tests) in order to adapt to the lower reflectivity of a human head (authors head used as reference). This gives the possibility to use a human head as object for position retrieval

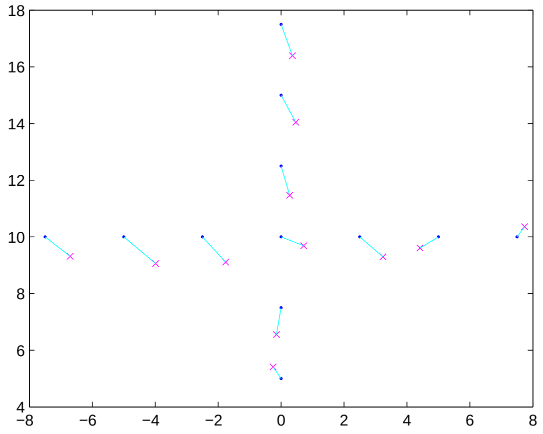


Figure 7.4: The 2D position sensor calibration, dots shows actual positions and x-marks retrieved positions after calibration. Units are cm.

measurements. This is done since the application of aim is a position detection of a human head.

7.3.1 Retrieving the Position of a Human Head

The measurements are made with a human head as the object, and the object of the 3D Reflection Map Model is a sphere of radius 9.5cm (approx. the size of the authors head), which moves on a plane surface. Figure 7.5 shows the results of a head position retrieval where the head movement done is: Center – right – left – center – forward – backward – center. The head movement is clearly visible on the figure. The measurements used here are obtained with acquisition rate $200Hz$, and are low pass filtered by taking average over 15 measurements to reduce the measurement noise. There are 100 retrieved positions in the plot, and the time used to retrieve these 100 positions is approx. 20.2ms using matlab (on a 3GHz Pentium). Acquisition time for 15 measurements is 75ms, which should make it possible to get a position acquisition rate of approx. $13Hz$.

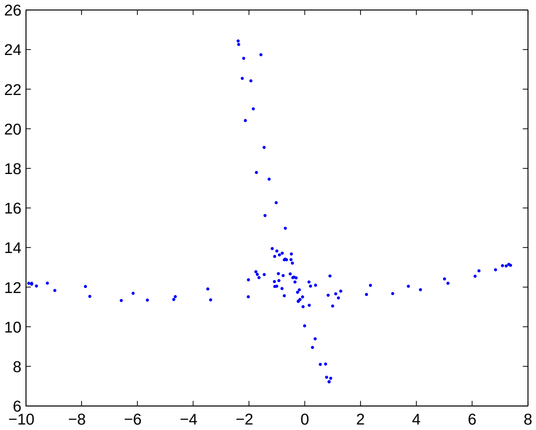


Figure 7.5: Positions retrieved from sphere movement.

7.4 2D Position Retrieval Method 2

The second method is based on direct calculations on the measured intensities. In Figure 7.6 nine reflections from the measurement sequence used to find the positions in Figure 7.5 are shown. Each reflection measurement is captioned with the position of the object (the head of the author) for that reflection. In the top left plot in Figure 7.6 the object is in center/neutral position. The top middle plot shows the reflections measured when

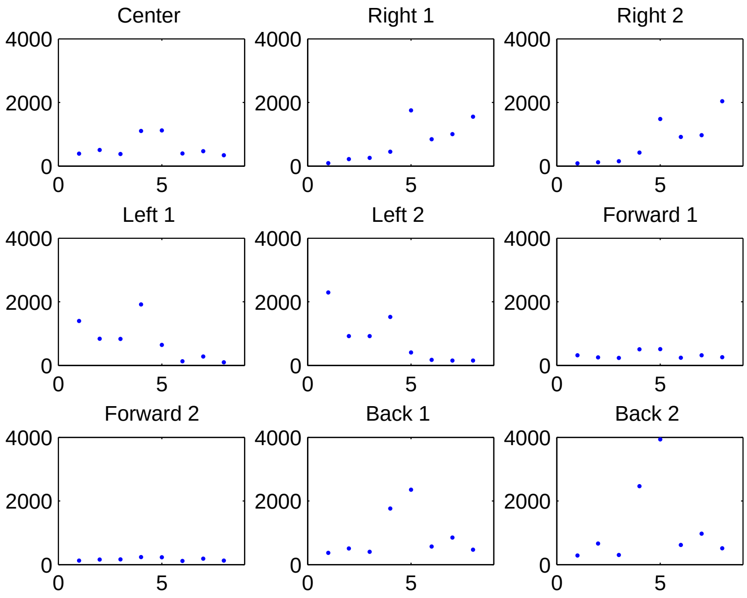


Figure 7.6: Reflections measured from object in positions as indicated by the sub-plot captions.

the object is moved a little to the right, and further movement of the object to the right results in reflections as shown on the top right plot. The same is trough for the remaining sub-plots of Figure 7.6, “left 1” means a little to the left and “left 2” means further away to the left.

From the sub-plots center, right 1, right 2, left 1 and left 2 it is obvious that the center of gravity in the plot moves. Also from the sub-plots center, forward 1, forward 2, back 1 and back 2 it can be seen that the center of gravity in the plot doen not move significantely, but the intensities varies depending on the distance from the object to the sensors. This can be exploited to get a measure of how the object is positioned.

First a reference measurement I_c is made with object in center position and a reference point for left–right movement is found as

$$x_\mathrm{c} = a\frac{w^\top I}{\sum I} \tag{7.1}$$

where the constant a is used to give the right scale.

Left–right movement is found by calculating the center of gravity of the measurements as

$$x = a\left(\frac{w^\top I}{\sum I} - x_\mathrm{c}\right) \tag{7.2}$$

where I is a column vector of measured intensities,
$w = [-3.5, -2.5, -1.5, -.5, .5, 1.5, 2.5, 3.5]^\top$ is the distances and x_c is the x coordinate from the reference reflection measurement.

The forward–backward movement if found by

$$y = -b\log\left(\frac{1}{4}\sum_{k=3}^{6}\frac{I_k}{I_{\mathrm{c}k}}\right) \tag{7.3}$$

where *log* is used for linerization, b scales to the right scale and I_c denotes the reference reflection for object in center position.

The scalars $a = 4$ and $b = 6$ in (7.1), (7.2) and (7.3) scales the estimated positions such that they become comparable to the positions from method 1, and so that the range are similar for left–right and forward–backward movements.

With this method object positions can be calculated as fast as the reflection measurements are made ($200Hz$ for this setup). A low pass filter to reduce the noise can the be applied to the retrieved positions, a filter that average over 10 positions seems reasonable.

Figure 7.7 shows the result of applying this "center of gravity" method to the same measurements as used to make Figure 7.5. Here all 3000 measurements are used to find positions and low pass filtering are then applied to the positions. The object movement in

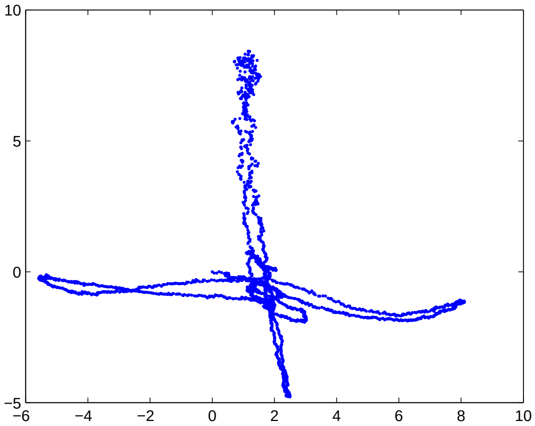


Figure 7.7: Positions retrieved from object movement using "center of gravity" method.

Figure 7.7 seems to be shifted some what to the right when compared to Figure 7.5, this is because the first measurement in the sequence is chosen as reference measurement, and the x coordinates of the first measurement in Figure 7.5 is -1.75cm.

7.4.1 Absolute Precision

Quantification of the absolute precision of the object position retrieval method 2 has been done in an indirect way. Because of problems with precise positioning of a human head, a set of retrieved position has been selected and then a camera has been used to picture of the head in that position. The absolute position of the human head is then found from the photo. Figure 7.8 shows a photo of the head positioned such that the retrieved position is $(0.0, 0.0)$. This position is chosen as origo for the absolute position of the human head.

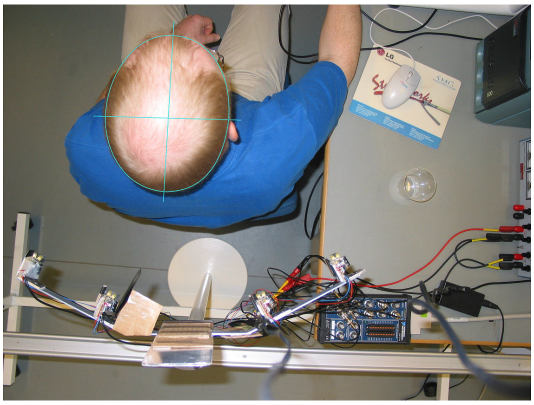


Figure 7.8: Photo of head in position for retrieving $(0, 0)$.

The ellipse with the cross-mark in Figure 7.8 indicates the center of the head which is the position that is retrieved. Figure 7.9 shows a photo of the head with the position-marker for retrieved position $(4, 0)$.

The pictures are taken with a digital camera at a fixed position aiming vertically down and using remote-capture software, such that the camera position is the same on all the pictures in the sequence. The absolute position of the head is then found by converting the pixel position of the center of the position marker to centimeters. No attempt to flatten the picture before the conversion has been made.

Figure 7.10 shows the actual head positions (dots) and (rescaled) retrieved positions (x-marks), pairwise connected by a line for trace purpose only. There is one measurement pair in Figure 7.10 that fall outside the majority, that is the measurement pair with the

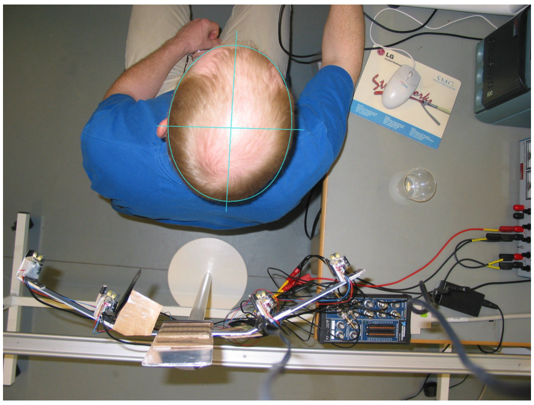


Figure 7.9: Photo of head in position for retrieving $(4, 0)$.

x-mark in position $(0, 3.4)$ and the dot in position $(-0.6, 4.8)$ a check measurement has shown that this measurement is an error, the retrieved position was not the expected one.

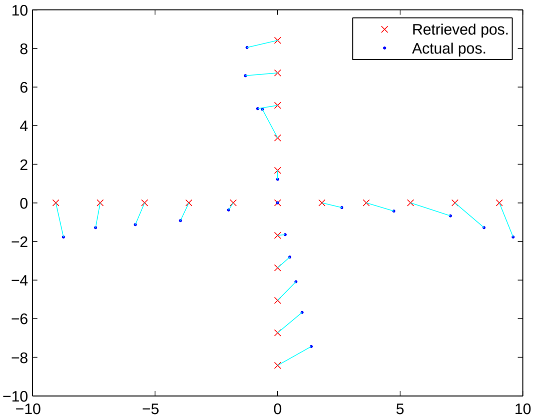


Figure 7.10: Actual head positions (dots) and (rescaled) retrieved positions (x-marks), pairwise connected by a line.

From Figure 7.10 it is found that the absolute accuracy of this method is less than 2cm, when the reference point (I_c) is set appropriate. If the reference point is chosen at a different position the result is not as good, this is shown in Figure 7.11. The measurements for this figure is not found with predetermined retrieved positions, but with a number of head positions, where the retrieved positions are recorded together with the photos of

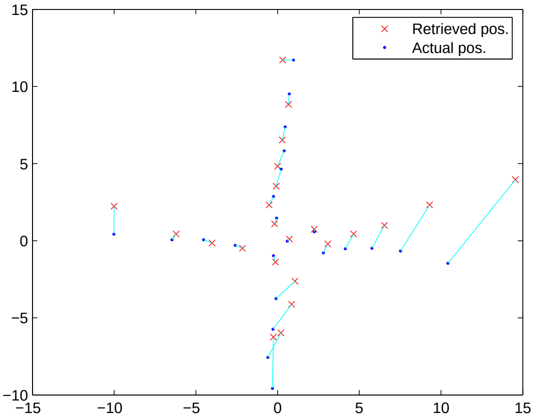


Figure 7.11: Actual head positions (dots) and (rescaled) retrieved positions (x-marks), pairwise connected by a line.

the head positions. For the retrieved position of the actual head position at $(-0.3, -9.6)$ the reason that it is close to the retrieved one for the actual head position $(-0.6, -7.5)$ is most likely that the receivers are being saturated.

It should be noticed here, that even though there is a great difference between the two calibration results in Figure 7.10 and Figure 7.11, the difference is almost not noticed when using the head position input device, as there is visual feedback from the dot on the screen.

7.5 Summary

The method using Nelder-Mead simplex algorithm for 3D position determination of Chapter 5 has been used to estimate positions constrained to two dimensionally space, with sensors places in the same plane. An other method for estimating two dimensional positions has been proposed, which does not use the 3D Reflection Map Model. This second 2D position Retrieval Method directly estimates the position from the measured intensities. The method is computationally much simpler than the one based on search trough the parameter space of the 3D Reflection Map Model, and the calibration of the second method can be done by one measurement, such a simple calibration has not been found for the other method. The precision of the position estimates of this second method has also been investigated.

Chapter 8

Realizing a Head Sensor for Controlling a Wheelchair

At my stay at Alcalá University it was proposed to me, that the light sensors that i worked on could be used for an input device for an electrical wheelchair. The input device should make it possible for a severely disabled person to control the movement of the wheelchair using head movements. At University of Alcalá the Department of Electronics have an electrically driven wheelchair used in their SIAMO project, which could be used for test of the input device, if the input device was ready for test before the end of my stay. The only condition for testing on their wheelchair was that I had to be the test driver. The input device reached a condition ready for a first test before the end of my stay in Alcalá, and the test drive showed some promising results.

The principle in the position detection of the head is the 2D Position Retrieval Method 2 from Chapter 7.

8.1 Introduction to Assistive Mobility

Several electrical powered wheelchairs to assist mobility of disabled persons are available with modular architecture e.g. [24] and The SIAMO project (Integral System for Assisted Mobility) [22]. The intention of this modular architecture is that it should be easy to configure the wheelchair to suite the needs of a large variety users with different disabilities. This modular architecture also makes it easy to adapt new functionality to the wheelchair, because e.g. the human-machine interface (HMI) can be changed, and new HMI can be made, without other changes to the wheelchair system [10].

For severe disabled persons one way of controlling a wheelchair is by use of head movements. There exists such devices today called *head controlled joystick* or *head-movement interface*, both mechanical, camera based [3], accelerometer based [29], and based on infrared light [8], where [29] and [8] use active components attached to the head of the user. The HMI proposed here is based on infrared light, and differs from [8] by the fact that no components are in physical contact with the head of the user.

The SIAMO wheelchair has an architecture where the HMI is independent from the rest of the system, such that the HMI can be changed without affecting the rest of the wheelchair system. This feature has been exploited to test the HMI proposed here on a real wheelchair.

8.2 The SIAMO Wheelchair

The SIAMO wheelchair was used for the first implementation and test of the "Head-Sensor", so a short description of the SIAMO project is given here. For further information on the SIAMO project see e.g. [22], [10], and [11].

SIAMO is the Spanish acronym for Integral System for Assisted Mobility. The SIAMO wheelchair is a prototype of an electrically driven wheelchair, which has a modular architecture such that a large variety of different human-machine interfaces (HMI) and/or environment perception modules. The SIAMO project was initiated in late 1996 at the Electronics Department of University of Alcalá, during this project the electronic system on the prototype wheelchair was entirely developed by the research team.

The system was designed to be versatile, such that it allows the incorporation or removal of various services by simply adding or removing the modules involved in the individual tasks. The main functionality blocks are:

a) Power and motion controllers
b) Human-Machine Interface (HMI)
c) Environment perception
d) Navigation and sensory integration

The power and motion controllers include the battery and charging system, and the propulsion motors and controllers for the these motors. The human-machine interface of HMI for short is the interface between the user and the chair. Environment perception is sensors that give information about the environment to the control system on the wheelchair. Navigation and sensory integration are the parts of the control system on the wheelchair that makes the decisions about the direction to move the wheelchair or

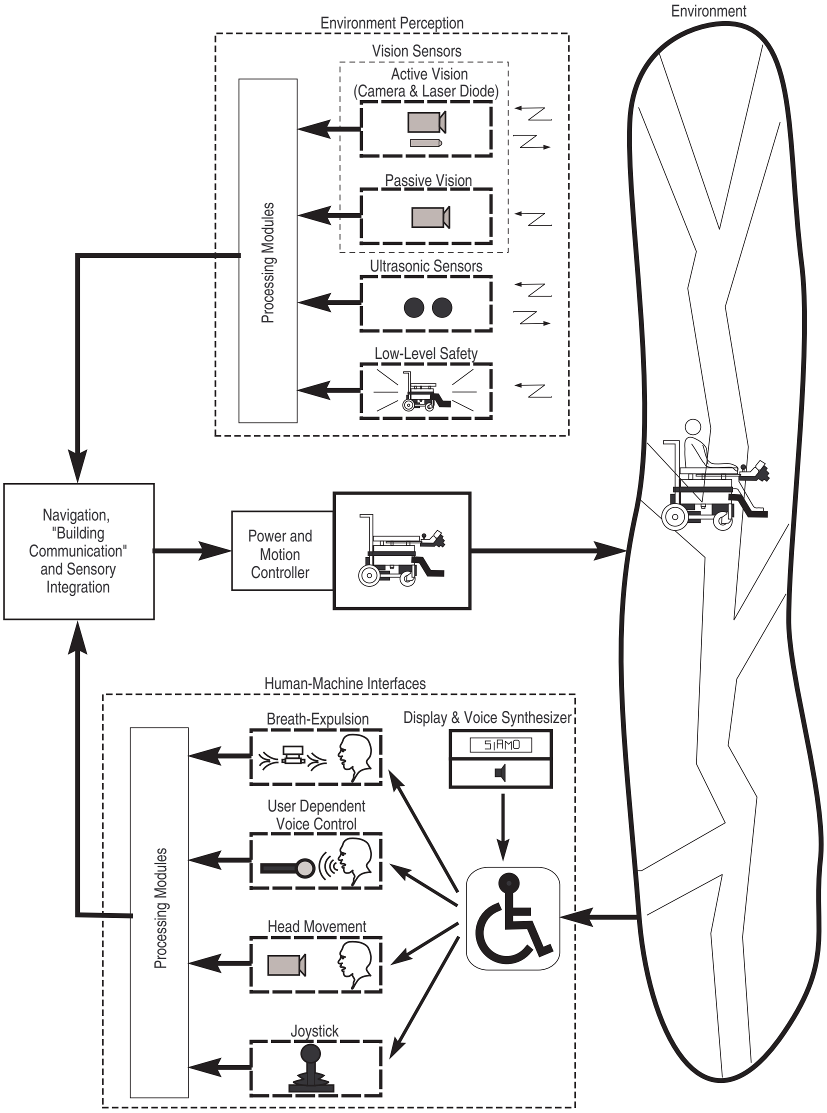


Figure 8.1: Block diagram of the system architecture of the SIAMO wheelchair.

not move the wheelchair depending on the input from the HMI and the environment perception. Figure 8.1 shows a block diagram of the system architecture of the SIAMO wheelchair, only some of the HMI's and environment perception modules are included in the block diagram.

Dependening on the user's needs and the characteristics of the environment, the wheelchair can be equipped with subsystems. Each subsystem can be reprogrammed to adapt it to special needs of the individual user. All subsystems of the SIAMO wheelchair communicates via a serial bus, so reconfiguration of the system is possible by adding or removing subsystems.

The configuration of the particular SIAMO wheelchair used for first test of the "Head-Sensor" was very basic, only the necessary power and motion controller, a linear joystick and the "Head-Sensor" HMI. The implementation of the "Head-Sensor" HMI was made on a battery powered PC, and the communication with the serial bus on the SIAMO wheelchair was made trough a generic subsystem connecting the parallel port of the PC to the serial bus of the SIAMO architecture.

8.3 Controlling an Electrical Wheelchair using Head Movements

Head position detection method 2 from Chapter 7 has been chosen for this implementation since it is the computationally simplest. This section contains details on how the setup on the SIAMO wheelchair, which was used for prototype tests.

8.3.1 Setup on the Wheelchair

The physical setup of sensors is not changed from the one used in Chapter 7. The sensors are placed behind the head of the wheelchair user, so that they do not limit the users field of view.

The sensors are grouped in two groups, one for each side of the object (head). Since no cross measurements from group to group are used it is possible to drive the emitters two by two on the same output signals, and there by only use two outputs from the computer. This has the advantage that one sampling cards available can handle the input and output from the computer. This solves the problem of having only one expansion slot in the computer available for tests on the wheelchair. The parallel driving of two emitters on the same output does influence the results from Chapter 7, since the physical placement of the sensors results in very limited cross-talk between the two groups of sensors.

Figure 8.2: The sensors attached to the head rest piece of the SIAMO wheelchair.

Figure 8.3: Photo of possible attachment of the disclosed HMI on a wheelchair.

Figure 8.2 shows the SIAMO wheelchair with the prototype of the new HMI attached to it. Figure 8.3 shows a close-up of the sensor arrangement. The computer that controls the sensors and performs the necessary processing of the measurements are placed behind the seat on the prototype, but the size of the computer allows for it to be placed under the seat. The communication network on the SIAMO wheelchair includes a parallel interface, which makes it possible to use the parallel port of a PC to communicate with the propulsion motors of the wheelchair.

The propulsion system on the wheelchair needs velocity commands for each of the two driving wheels in radians per second. Rather than specify the speed for the left and right driving wheels individually, the linear and angular velocity of the wheelchair is selected as the output of the head movement to wheelchair movement converter, described in the next subsection. The translation from linear velocity v and angular velocity ω to velocities for the left and right driving wheels of the wheelchair is:

$$\omega_{\mathrm{r}} = \frac{v + \omega\frac{d}{2}}{r}$$
$$\omega_{\mathrm{l}} = \frac{v - \omega\frac{d}{2}}{r}$$

where D is distance between the two driving wheels of the wheelchair, and R is the radius of these wheels.

8.3.2 Conversion of Head Movements to Wheelchair Movements

In order to control the wheelchair the head positions have to be converted to movement orders for the wheelchair propulsion motors. For this purpose the "area" of possible head positions was at first divided into nine sectors as shown in Figure 8.4 where measured head head positions are superimposed into the sectioning. The head position measurements are the same as used in Figure 7.7 in Chapter 7, but with a different measurement used as center reference (measurement no. 175). As seen in Figure 8.4 the resolution of head positions is sufficient for nine sectors which can be used for driving commands.

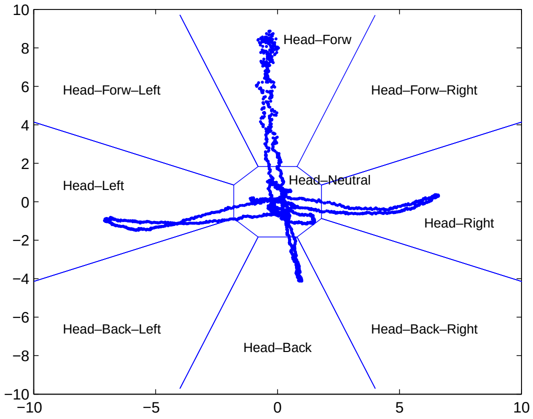


Figure 8.4: Retrieved Head positions and the nine sectors.

Table 8.1 shows the conversion table for the head movement to wheelchair movement converter. The number of commands has been kept at a minimum, to keep implementation simple. The commands in Table 8.1 are useful to guide the wheelchair, but for a commercial version of the head sensor device they may not be sufficient. Especially the

User Action	Driving Command
Head Forward	Increase Speed
Head Backward	Decrease Speed
Head Centered	Continue at Present Speed
Head Left	Turn Left
Head Right	Turn Right
Quick Backw.-Forw.	Stop Forward Motion
Quick Forw.-Backw.	Stop Backward Motion

Table 8.1: Conversion table for the head movement to wheelchair movement converter.

codification of the stopping commands could be discussed, and the lack of commands for turning the sensor on and off might also be advantageously.

The commands are divided into two groups: Direct commands which are the first five in Table 8.1, and codified commands which are the two commands for stopping forward and backward motion respectively. With this limited number of commands the number of sections used in the converter has been reduced to five, as shown in Figure 8.5. The sector Neutral is a circle of radius two.

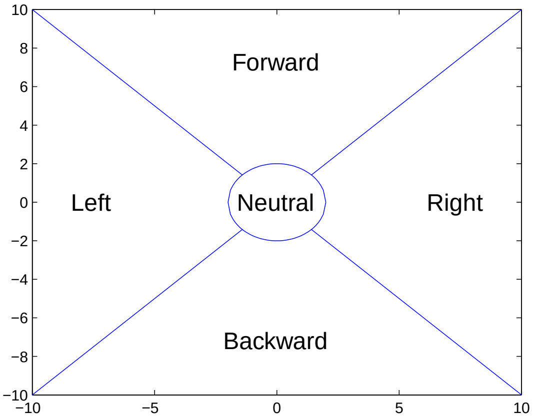


Figure 8.5: Area of possible head positions divided into five sectors.

A state machine is used to control the linear speed (forward/backward speed) of the wheelchair, based on the section in which the head currently is. Figure 8.6 shows the state machine for the normal increase/decrease of the velocity. The variables in the state machine are linear velocity v, linear speed increment/decrement Δv, and time step Δt.

To stop forward/backward motion instantaneously codification is used, so that a movement of the head in the opposite direction and quickly back to neutral will stop the

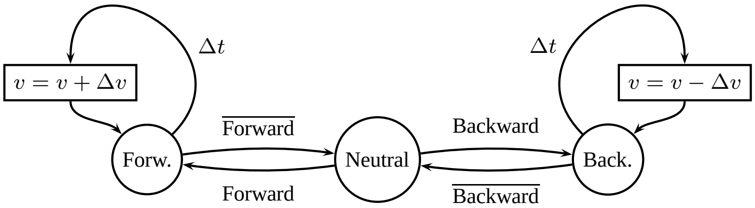


Figure 8.6: State machine for converting head positions to linear speed.

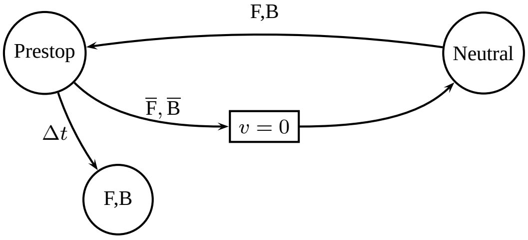


Figure 8.7: State machine part for codified stop commands.

wheelchair. If the head is not moved back to neutral position quickly the result is normal increase/decrease of the velocity according to head movement. Figure 8.7 shows the state machine part that takes care of the codified stopping commands. F and B in the figure references to Forward and Backward, respectively.

The time for a quick move is set to $\Delta t \leq 0.3$s and normal increase/decrease of linear speed will be in steps of $\Delta v = 0.05$m/s every half a second with the first increment/decrement after $\Delta t = 0.3$s to lower the respond time. These values are subject to change to fit the needs and abilities for the individual user.

Turning the wheelchair was initially intended to follow a similar scheme, with start and stop of angular motion. It turned out to be very difficult to control the wheelchair using this control scheme. Instead the angular velocity is controlled directly by the user by the degree of head tilt. Little tilt of the head results in low turning speed, and more tilt of the head raises the turning speed. two additional steps have been made to make it easier to drive straight ahead without sway: A dead zone in the center position and making the angular speed partly inverse proportional to the linear speed.

$$\omega = -\frac{x}{\alpha + \beta|v|} \; (|x| > 1)$$

where x is the x coordinate of the actual head position (measuring head tilt), the constant α controls the degree of head tilt resulting in maximum angular speed, and β is the proportional factor for inverse proportionality to linear speed. In the test setup the used are $\alpha = 5$ and $\beta = 30$.

Finally the frequency of supplying speed commands for the wheelchair propulsion motors, was initially set too high, with the result that the wheelchair showed slow reaction times. The reason was that the speed commands queued up, and processing time resulted in a delay. Decreasing the command frequency to about ten commands a second eliminated this problem.

8.4 User Safety

The purpose of a wheelchair is to assist a human being in mobility, so the safety of that human need some consideration. As mentioned above codification can be used such that certain movements from the user will stop the motion of the wheelchair. In an emergency case such codified movements might not be the best way to stop the wheelchair. Either because it might take too long, or because panic might prevent the wheelchair user from doing the codified movement correctly.

The algorithm used to give driving commands to the wheelchair propulsion motors can be implemented in such a way that only a limited part of the full motion range of the head is used for driving control. If the head is moved outside this range for normal wheelchair driving control the commands to the wheelchair can be stop immediately (Emergency stop). The possibility to combine the presented HMI with a more traditional emergency stop button is also available. This emergency stop button could e.g. be placed behind the head in the outer limit of the head movement area, such that in an emergency situation the user can push the button by bending the head all way back.

In the present implementation the emergency stop function for head moved outside a limited part of the full motion range of the head is used.

Also the maximal speed of the wheelchair was limited to 2m/s for linear motion and 2rad/s for angular motion. This is for safety of the wheelchair test driver, for the untrained wheelchair driver (such as the author) 2m/s feels very fast.

8.5 Test and Test Drives with the Wheelchair

The tests and test drives described here has the author as test driver unless otherwise is stated.

For the first tests a simple wheelchair simulator was implemented. The simulator included the whole setup as made for the wheelchair, but instead of feeding the wheel velocities to the wheelchair propulsion motor controllers, a virtual wheelchair was moved on the computer screen. This wheelchair simulator served two purposes, it was used to test the functionality of the software prior to test drives with the SIAMO wheelchair, and for training in using the input device.

After test on the wheelchair simulator the SIAMO wheelchair was equipped with the new HMI. Several test drives have been made, some of which raised the need for changes in the control scheme, as indicated in the description of the head movement to wheelchair movement converter of Section 8.3.

A first test drive in a hall area showed that the control of the wheelchair is fairly easy and feels natural to a first time wheelchair user.

A second test drive showed that it was possible to do small corrections to the driving direction as the wheelchair was driven down a corridor and trough a doorway. This test was carried out with two different drivers, besides the author, Juan Carlos García was test driver on a drive down the corridor. This showed that the HMI is robust against changes in head size and hair color, as the author has blond hair and Juan Carlos has black hair. No recalibration was needed when changing driver.

Finally, an attempt to drive the wheelchair into an elevator was successful and showed that it is possible to use the proposed HMI to navigate into narrow places.

Under one of the test drives the wheelchair was driven past a number of windows, and the head position detection was influenced severely by the sun light shining trough the windows. The functionality of the head sensor was not maintained during the pass, but the robustness against light from surroundings e.g. sun light directly on the receivers needs further work.

8.6 Summary

A new infrared non-contact head sensor has been proposed for HMI to wheelchair control. The HMI has been tested both in laboratory setup and on an actual wheelchair prototype. A wheelchair simulator has also been implemented, and tests using this simulator have shown that the HMI has robust performance, both to variations in the sensors and to the changing hair color of different users.

The proposed HMI has been implemented on the SIAMO wheelchair as a new input device module. Test drives with the wheelchair have shown that proposed HMI is useful for controlling the wheelchair, and that the control of the wheelchair feels natural and comfortable. Also, the commands and behavior of the sensor can be easily reprogrammed, adapting it to other ways of driving or specific constrains of some kind of potential users.

One thing that need more work is robustness against light from surroundings e.g. sun light directly on the receivers.

Chapter 9

Robot Orientation Sensor

In this chapter the 3D Reflection Map Model from Chapter 4 is adapted to model reflections from environment prototypes for obstacles a robot encounters when driving in an indoor environment. The aim of the model adaption is to develop an orientation sensor for use on an indoor autonomous robot, so that it can detect the type, range and orientation of obstacles in the operating environment.

9.1 Model of Environment Primitives

With the aim of making an orientation sensor for a robot, the 3D Reflection Map Model from Chapter 4 is modified to model reflections from the primitives plane, edge, corner and cylinder. Figure 9.1 shows the four primitives for which the reflected light has to be modeled.

The reflection models for the primitives in Figure 9.1 contains the same elements as the 3D Reflection Map Model with the only difference that the object is now one of the

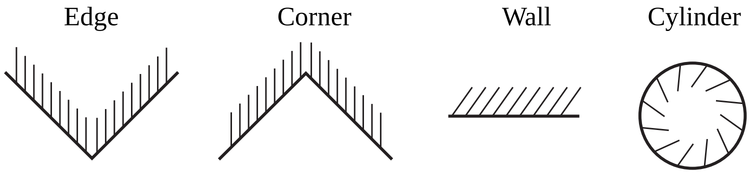


Figure 9.1: The four primitives Corner, Edge, Plane, and Cylinder, for which the light reflection will be modeled.

primitives in place of the sphere.

First the model is modified to model the reflections from a corner, wall or edge. All these three primitives can be modeled by the same reflection model. To model the reflections from a cylinder an other modification of the 3D Reflection Map Model is needed.

9.1.1 Reflection Model for Edge, Corner, and Plane

The primitives edge, corner is modeled as two half planes, with one point in common (the edge/corner point), they can be described by three parameters: The coordinate of the point and two normal vectors, one for each half plane.

The plane can be modeled by the same setup, when the two normal vectors have the same direction. This might seem to be a rather complicated way of modeling a plane, but it has the advantage that it is a special case of the edge/corner model.

Let the edge primitive be the choice of object for development of the model. The setup of emitter, receiver and corner is given by: Emitter position, E, emitter orientation, $\boldsymbol{n}_{\mathrm{E}}$, receiver position, R, receiver orientation, $\boldsymbol{n}_{\mathrm{R}}$, edge coordinate, $\boldsymbol{P}_{\mathrm{c}}$, and normal vectors for wall one and two, $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$, respectively. This is illustrated in Figure 9.2.

For simplicity assume that the emitter is at origo, and that the receiver is on the positive half of the y axis. This can always be obtained by translation and rotation as described in Subsection 4.2.1 on page 27.

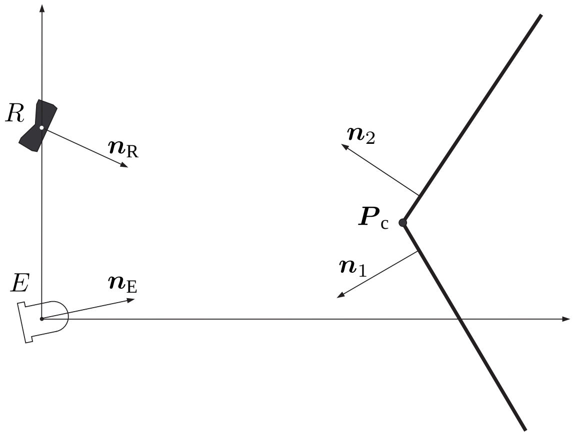


Figure 9.2: The setup for an edge, including emitter, receiver and edge.

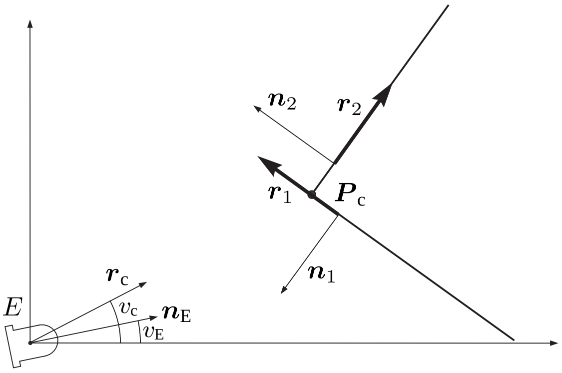


Figure 9.3: Direction vectors and angles.

The angle of direction of the emitter is

$$\upsilon_\mathrm{E} = \arctan\left(\frac{\boldsymbol{n}_\mathrm{E}}{\|\boldsymbol{n}_\mathrm{E}\|}\right)$$

The direction vector, $\boldsymbol{r}_\mathrm{c}$, and angle, υ_c, from the emitter (origo) to the corner point is found as

$$\begin{aligned}\boldsymbol{r}_\mathrm{c} &= \frac{\boldsymbol{P}_\mathrm{c}}{\|\boldsymbol{P}_\mathrm{c}\|}\\ \upsilon_\mathrm{c} &= \arctan\left(\frac{\boldsymbol{r}_\mathrm{cy}}{\boldsymbol{r}_\mathrm{cx}}\right)\end{aligned}$$

To determine if light in a certain direction from the emitter hit one of the walls, the direction vectors of the walls are found as

$$\begin{aligned}\boldsymbol{r}_1 &= \begin{bmatrix}\boldsymbol{n}_\mathrm{1y} & -\boldsymbol{n}_\mathrm{1x} & 0\end{bmatrix}^\top\\ \boldsymbol{r}_2 &= \begin{bmatrix}\boldsymbol{n}_\mathrm{2y} & -\boldsymbol{n}_\mathrm{2x} & 0\end{bmatrix}^\top\end{aligned}$$

Then the projection of the direction vector for the emitted light onto the line trough $\boldsymbol{P}_\mathrm{c}$ with direction $\boldsymbol{r}_1$ will be positive when the light hits wall one. Similarly for wall two.

Figure 9.3 shows the direction vectors $\boldsymbol{r}_\mathrm{c}$, $\boldsymbol{r}_1$, $\boldsymbol{r}_2$ and angles υ_E and υ_c.

The integration limits for the ν and θ integrals are chosen to be $\pm 65^\circ$ equal to the half angle of the particular emitter in use. The center of both integral limits are the emitter direction, $\boldsymbol{n}_\mathrm{E}$ (which is assumed to be contained in the xy plane).

Let the angle of direction for light emitted from the emitter be denoted by θ then the direction vector is

$$\boldsymbol{r}_\mathrm{P} = \begin{bmatrix}\cos(\upsilon_\mathrm{E} + \theta) & \sin(\upsilon_\mathrm{E} + \theta) & 0\end{bmatrix}^\top$$

The line from origo with direction $\boldsymbol{r}_{\mathrm{P}}$ has a normal vector

$$\boldsymbol{n}_{\mathrm{P}} = \begin{bmatrix} -\boldsymbol{r}_{\mathrm{Py}} & \boldsymbol{r}_{\mathrm{Px}} & 0 \end{bmatrix}^{\top}$$

Now the condition for the light beam from the emitter (at origo) with direction $\boldsymbol{r}_{\mathrm{P}}$ to hit the front side of wall one is

$$-\boldsymbol{r}_{\mathrm{P}}^{\top}\boldsymbol{n}_1 > 0$$

The point P on wall one is the point where the line of emitted light intersects the line of wall one (the point hit by the light beam). The line of emitted light has the equation

$$\boldsymbol{n}_{\mathrm{Px}}(x - x_0) + \boldsymbol{n}_{\mathrm{Py}}(y - y_0) = 0 \quad \Leftrightarrow \quad \boldsymbol{n}_{\mathrm{Px}}x + \boldsymbol{n}_{\mathrm{Py}}y = 0$$

since the emitter is positioned in origo. An equation for the wall one line is

$$\boldsymbol{n}_{1\mathrm{x}}(x - x_0) + \boldsymbol{n}_{1\mathrm{y}}(y - y_0) = 0 \quad \Leftrightarrow \quad \boldsymbol{n}_{1\mathrm{x}}x + \boldsymbol{n}_{1\mathrm{y}}y = \boldsymbol{n}_{1\mathrm{x}}\boldsymbol{P}_{\mathrm{cx}} + \boldsymbol{n}_{1\mathrm{y}}\boldsymbol{P}_{\mathrm{cy}}$$

since $\boldsymbol{P}_{\mathrm{c}}$ is a point on wall one.

Now if the two lines are parallel then there does not exists an intersection point. An intersection point exists when the normal vectors $\boldsymbol{n}_{\mathrm{P}}$ and $\boldsymbol{n}_1$ are not in parallel

$$\begin{vmatrix} \boldsymbol{n}_{\mathrm{Px}} & \boldsymbol{n}_{\mathrm{Py}} \\ \boldsymbol{n}_{1\mathrm{x}} & \boldsymbol{n}_{1\mathrm{y}} \end{vmatrix} <> 0$$

For the situation where there is not an intersection point no light is reflected. In case of an intersection point the x and y coordinate of the intersection point is found as

$$\boldsymbol{P}_{\mathrm{xy}} = \left[\frac{\begin{vmatrix} 0 & \boldsymbol{n}_{\mathrm{Py}} \\ c_1 & \boldsymbol{n}_{1\mathrm{y}} \end{vmatrix}}{\begin{vmatrix} \boldsymbol{n}_{\mathrm{Px}} & \boldsymbol{n}_{\mathrm{Py}} \\ \boldsymbol{n}_{1\mathrm{x}} & \boldsymbol{n}_{1\mathrm{y}} \end{vmatrix}} \quad \frac{\begin{vmatrix} \boldsymbol{n}_{\mathrm{Px}} & 0 \\ \boldsymbol{n}_{1\mathrm{x}} & c_1 \end{vmatrix}}{\begin{vmatrix} \boldsymbol{n}_{\mathrm{Px}} & \boldsymbol{n}_{\mathrm{Py}} \\ \boldsymbol{n}_{1\mathrm{x}} & \boldsymbol{n}_{1\mathrm{y}} \end{vmatrix}} \quad 0 \right]^{\top}$$

where $c_1 = \boldsymbol{n}_{1\mathrm{x}}\boldsymbol{P}_{\mathrm{cx}} + \boldsymbol{n}_{1\mathrm{y}}\boldsymbol{P}_{\mathrm{cy}}$. Now the only thing to test is that the point P is on the right side of $\boldsymbol{P}_{\mathrm{c}}$ i.e. on the half line that represents the wall. This is the case when the direction vector of wall one $\boldsymbol{r}_1$ and the vector from $\boldsymbol{P}_{\mathrm{xy}}$ to $\boldsymbol{P}_{\mathrm{c}}$ has the same direction. Then the inner product of the two vectors is non negative:

$$(\boldsymbol{P}_{\mathrm{c}} - \boldsymbol{P}_{\mathrm{xy}})^{\top}\boldsymbol{r}_1 \geq 0$$

Figure 9.4 shows a situation where the light beam from the emitter intersects the line representing the wall on the wrong side of $\boldsymbol{P}_{\mathrm{c}}$. This is a false solution because there are no wall at the intersection point.

The normal vector of the point $\boldsymbol{P}$ equals $\boldsymbol{n}_1$.

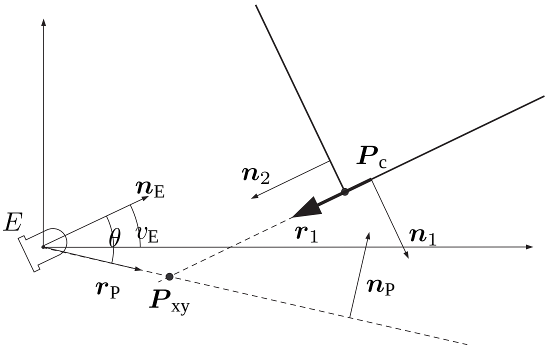


Figure 9.4: Sketch of situation with a false solution.

Similarly for wall two the condition for the light beam hitting the front side of the wall is

$$-\boldsymbol{r}_\text{P}^\top \boldsymbol{n}_2 > 0$$

The line of wall two are

$$\boldsymbol{n}_\text{2x}(x - x_0) + \boldsymbol{n}_\text{1y}(y - y_0) = 0 \quad \Leftrightarrow \quad \boldsymbol{n}_\text{2x}x + \boldsymbol{n}_\text{2y}y = \boldsymbol{n}_\text{2x}\boldsymbol{P}_\text{cx} + \boldsymbol{n}_\text{2y}\boldsymbol{P}_\text{cy}$$

The condition for an intersection point is

$$\begin{vmatrix} \boldsymbol{n}_\text{Px} & \boldsymbol{n}_\text{Py} \\ \boldsymbol{n}_\text{2x} & \boldsymbol{n}_\text{2y} \end{vmatrix} <> 0$$

And the x and y coordinates of $\boldsymbol{P}$ is

$$\boldsymbol{P}_\text{xy} = \begin{bmatrix} \dfrac{\begin{vmatrix} 0 & \boldsymbol{n}_\text{Py} \\ c_2 & \boldsymbol{n}_\text{2y} \end{vmatrix}}{\begin{vmatrix} \boldsymbol{n}_\text{Px} & \boldsymbol{n}_\text{Py} \\ \boldsymbol{n}_\text{2x} & \boldsymbol{n}_\text{2y} \end{vmatrix}} & \dfrac{\begin{vmatrix} \boldsymbol{n}_\text{Px} & 0 \\ \boldsymbol{n}_\text{2x} & c_2 \end{vmatrix}}{\begin{vmatrix} \boldsymbol{n}_\text{Px} & \boldsymbol{n}_\text{Py} \\ \boldsymbol{n}_\text{2x} & \boldsymbol{n}_\text{2y} \end{vmatrix}} & 0 \end{bmatrix}^\top$$

where $c_2 = \boldsymbol{n}_\text{2x}\boldsymbol{P}_\text{cx} + \boldsymbol{n}_\text{2y}\boldsymbol{P}_\text{cy}$. Finally the point $\boldsymbol{P}$ is on wall two when the vector from $\boldsymbol{P}_\text{xy}$ to $\boldsymbol{P}_\text{c}$ has opposite directions:

$$(\boldsymbol{P}_\text{xy} - \boldsymbol{P}_\text{c})^\top \boldsymbol{r}_1 \geq 0$$

Here the normal vector of the point $\boldsymbol{P}$ equals $\boldsymbol{n}_2$.

The z coordinate of $\boldsymbol{P}$ is found from ν as

$$\boldsymbol{P}_\text{z} = \|\boldsymbol{P}_\text{xy}\| \arctan(\nu)$$

From here the rest of the model is the same as the 3D Reflection Map Model of Chapter 4, starting from equation (4.4) on page 33.

9.1.2 Reflection Model for Cylinder

The cylinder in this reflection model is assumed to be vertical. The reflection model for the vertical cylinder is quite similar to the 3D Reflection Map Model for the sphere in Chapter 4. The cylinder is described by the center position in the xy plane $\boldsymbol{C}$ and the radius r. The first part of this model follows the 3D Reflection Map Model as far as determination of the points q_1 to q_4, i.e. equation (4.3). See Figure 4.3 on page 29 for the position of the points q_1 to q_4.

The integration limits for the θ integral is then found as

$$\begin{aligned}\theta_1 &= \arctan\left(\frac{q_{3\mathrm{y}}}{q_{3\mathrm{x}}}\right)\\ \theta_2 &= \arctan(\alpha_2)\end{aligned}$$

where α_2 is the slope of the line from the emitter that tangent the circle (projection of the cylinder onto the xy plane) as found in Equation (4.2) on page 29.

The direction in the xy plane of light emitted by the emitter is denoted θ, and the angle to the xy plane is denoted by ν. Denoting the distance from the emitter (origo) to the intersection point of the line of emitted light and the circle by t_p the intersection point between the line of light and the circle is found as

$$\boldsymbol{P} = t_p \begin{bmatrix}\cos(\theta) & \sin(\theta) & \tan(\nu)\sqrt{1 + tan^2(\theta)}\end{bmatrix}^\top$$

where

$$t_p = \boldsymbol{C}_\mathrm{x}\cos(\theta) + \boldsymbol{C}_\mathrm{y}\sin(\theta) - \sqrt{(\boldsymbol{C}_\mathrm{x}\cos(\theta) + \boldsymbol{C}_\mathrm{y}\sin(\theta))^2 + r^2 - \boldsymbol{C}_\mathrm{x}^2 - \boldsymbol{C}_\mathrm{y}^2}$$

Now the unit normal vector $\boldsymbol{n}_\mathrm{P}$ to the cylinder at point $\boldsymbol{P}$ can be found as

$$\boldsymbol{n}_\mathrm{P} = \frac{\begin{bmatrix}\boldsymbol{P}_\mathrm{x} - \boldsymbol{C}_\mathrm{x} & \boldsymbol{P}_\mathrm{y} - \boldsymbol{C}_\mathrm{y} & 0\end{bmatrix}^\top}{\left\|\begin{bmatrix}\boldsymbol{P}_\mathrm{x} - \boldsymbol{C}_\mathrm{x} & \boldsymbol{P}_\mathrm{y} - \boldsymbol{C}_\mathrm{y} & 0\end{bmatrix}^\top\right\|}$$

The rest of the model follows the 3D Reflection Map Model of Chapter 4 from Equation (4.4).

9.2 Model Validation

The modified models for modeling the reflections from the four primitives wall, corner, edge, and cylinder are validated by comparing the modeled reflections with measured reflections for the setup shown in Figure 9.5 for the corner.

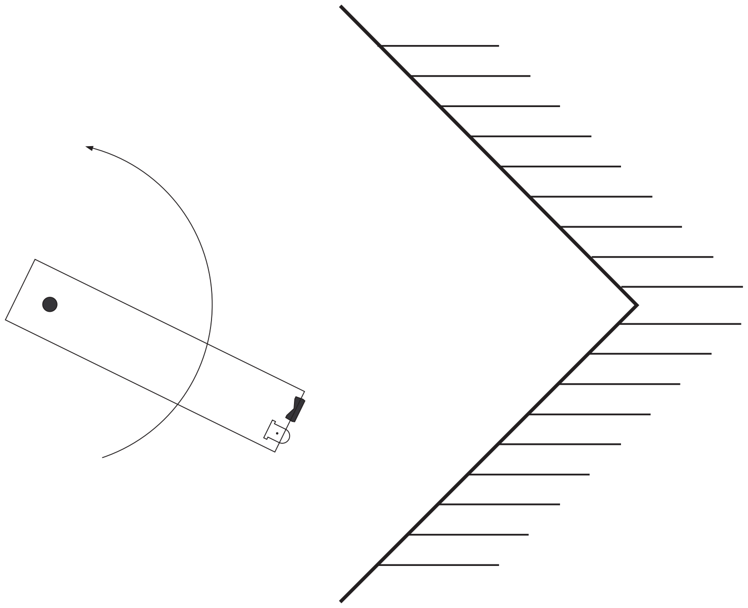


Figure 9.5: The setup used for validating the models for reflections from wall, corner (shown), edge and cylinder.

The measurements are made for a number of distances from emitter/receiver pair to the object. For the corner and edge the distances are measured to the corner/edge point, and the center of the rotation of the emitter/receiver pair is placed on the half angle line for the corner or edge. The radius of rotation for the emitter/receiver pair is 17cm. The wall, corner and edge used in the model validation were white painted walls in the laboratory area. The cylinder in use was a white cardboard tube of radius 3.6cm.

Figure 9.6–9.9 shows the results of the results of the model validations.

Figure 9.6 and Figure 9.7 shows that the modeled and measured intensity of reflected light are in good accordance for both these cases.

Figure 9.8 and Figure 9.9 shows some deviation between modeled and measured intensity of reflected light. By changing the emitter characteristic in the model from cosine to uniform the deviation between modeled and measured reflections are almost eliminated, as shown in Figure 9.10 for the corner and Figure 9.11 for the cylinder.

In the corner case the reason for the uniform emitter characteristic gives a better result might be that there are unmodeled multi-paths reflections that gives a small contribution to the measurements.

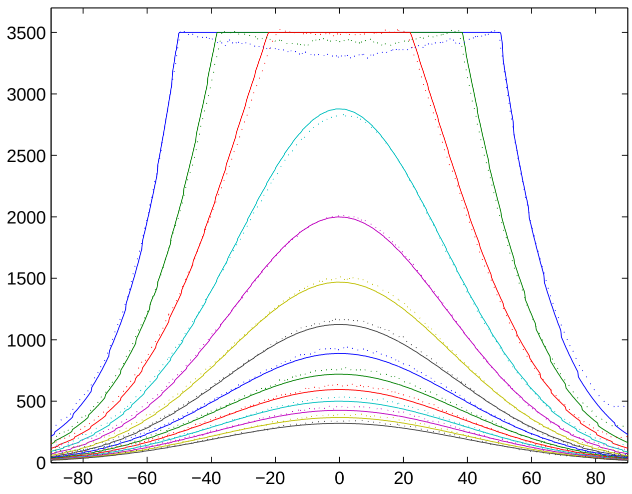


Figure 9.6: Model validation for wall reflection model.

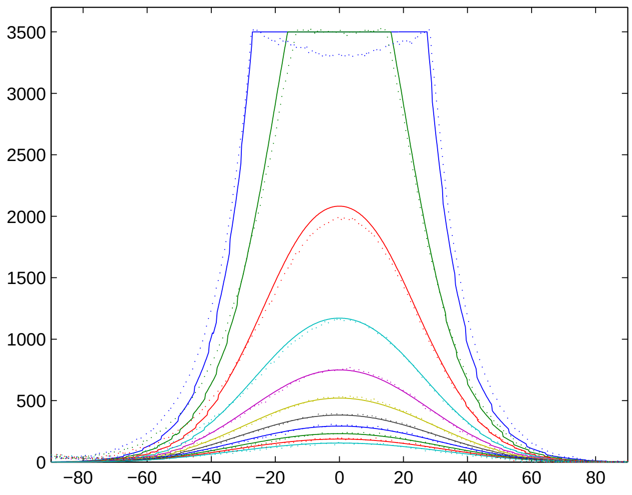


Figure 9.7: Model validation for edge reflection model.

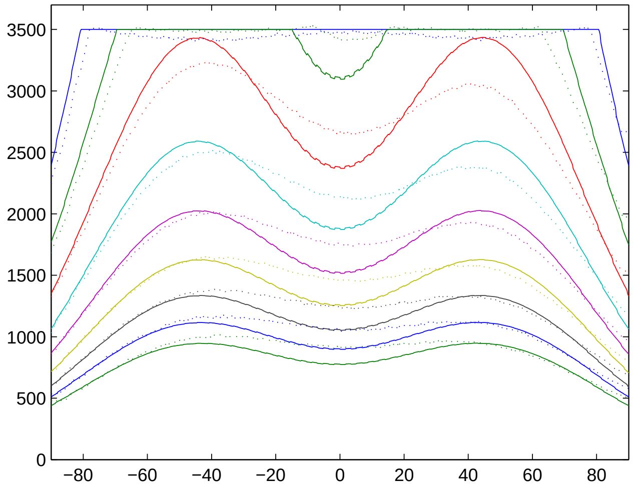


Figure 9.8: Model validation for corner reflection model.

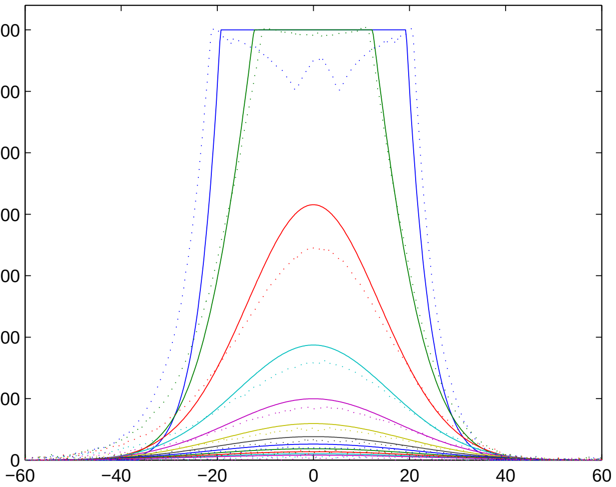


Figure 9.9: Model validation for cylinder reflection model.

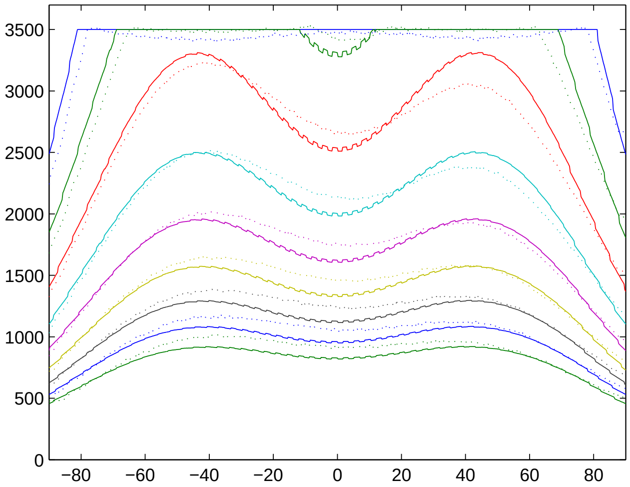


Figure 9.10: Model validation for corner reflection model using uniform emitter characteristic.

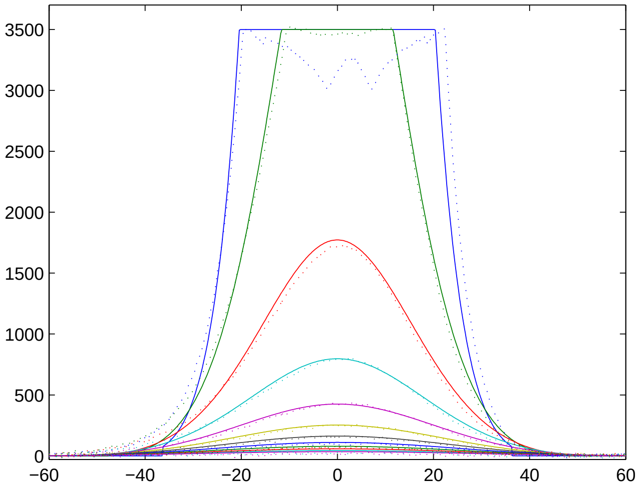


Figure 9.11: Model validation for cylinder reflection model using uniform emitter characteristic.

9.3 Orientation Estimation by COG method and Polynomial fit based method

The investigation in this section has only been done on modeled data from the reflection models. Further investigation and especially tests on real measurements are necessary. The experiments with modeled data gives some interesting results.

The same modeled reflection data as used in the preceding section has been investigated for possibilities of using the same principle with center of gravity as used for position determination of the human head in Section 7.4, for determination of orientation of the four prototypes plane, edge, corner and cylinder.

First a number of reflections for different angles and distances have been plotted for each of the four primitives, see Figure 9.12 and Figure 9.13.

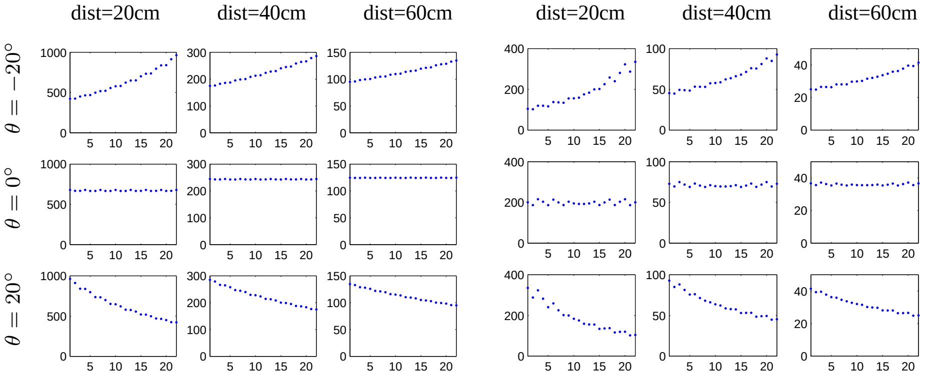


Figure 9.12: Modeled reflections from a plane, left, and an edge, right, at different distances and angles.

The modeled reflections for both plane and edge in Figure 9.12, shows a pattern that changes with changing angle, but are almost constant for changing distances. The modeled reflections from the corner shows almost no changing except falling amplitude with growing distance, and for the cylinder the changes depends on both angle and distance.

For the cylinder the principle from Section 7.4 may be applied (with adapted weights w and scales a and b) to give an estimate of the 2-dimensional position of the cylinder in front of the sensor setup. Figure 9.14 shows the result of applying this principle to the modeled reflections from a cylinder. The actual positions is marked by a dot and the x-marks are the retrieved positions.

As seen in Figure 9.14 the area where retrieved and actual positions are close to each other is rather limited. The conclusion of this is that the COG method is only applicable

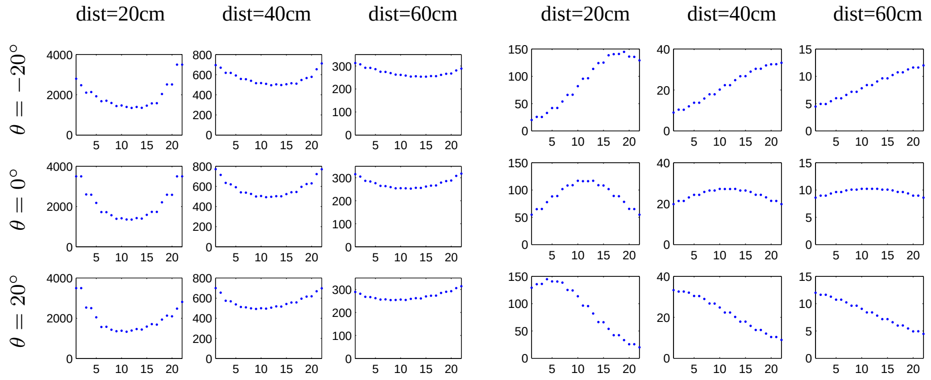


Figure 9.13: Modeled reflections from a corner, left, and a cylinder, right, at different distances and angles.

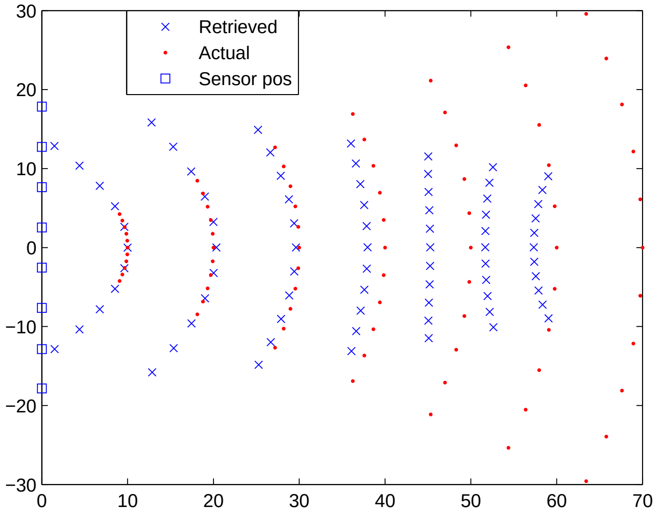


Figure 9.14: The COG position determination method applied to modeled reflections from a cylinder.

when the area of where position determination has to be done is small. This was indeed the case with the head sensor in Chapter 8.

Note, that the reference for the position used here is the reflections from position $(10, 0)$. The reference position will always coincide with the estimate of the reference position (by construction of the method). This can be utilized if the aim is to keep the cylinder in a certain position using the reflections from this position as the reference.

In the case of planes the orientation of planes it has been found that the orientation of the plane can be estimated by

$$\theta_{\mathrm{p}} = c\frac{\boldsymbol{w}^{\top}\log(\boldsymbol{I}_{\mathrm{p}})}{\sqrt{\sum \boldsymbol{I}_{\mathrm{p}i}}} \tag{9.1}$$

where $\boldsymbol{w} = \begin{bmatrix} -10.5 & -9.5 & -8.5 & \dots & 9.5 & 10.5 \end{bmatrix}^{\top}$, and c is a scalar. Figure 9.15 shows the actual (solid) and estimated (dotted) orientation angle for modeled reflections from a plane in the range $\pm 25°$.

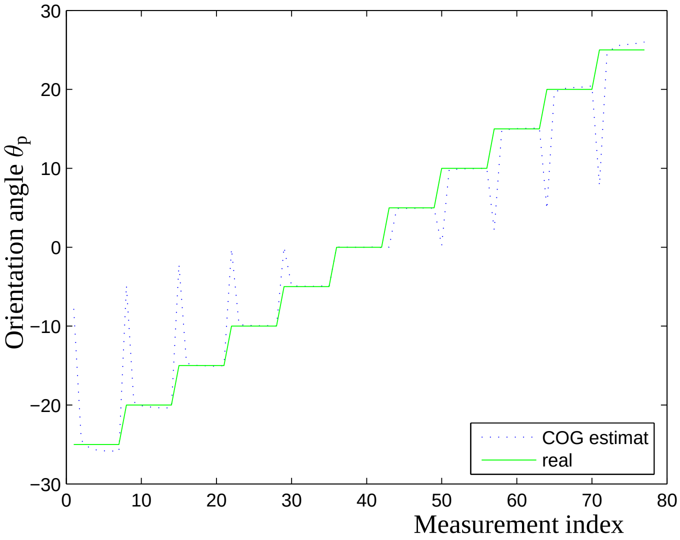


Figure 9.15: The COG orientation estimate for planes.

The large peak-like differences in Figure 9.15 originates from the fact that when the plane is placed closest to the sensors some of the receivers are saturated. This saturation of some of the receivers results in a faulty estimate of the orientation of the wall. If this method of estimating the orientation of a plane is to be implemented at some later time, it is possible to detect the saturation of receivers in order to invoke some appropriate special action.

Orientation of edges can also be estimated by Equation 9.1. The result of this is shown in Figure 9.16, where it can be seen that there are also some spike-like differences between actual and estimated orientation of the edge. For the edge case there is no saturation of receivers when the edge is placed close to the sensors, so an alternative method for estimating the orientation of the edge has been found, based on fitting the light intensities of the reflections with a polynomial of degree two.

The polynomial fit estimation procedure applied to the reflections from the edge is:

$$\begin{aligned} P &= \begin{bmatrix} \boldsymbol{w}^2 & \boldsymbol{w} & \mathbf{1} \end{bmatrix} \\ \boldsymbol{a} &= (P^\top P)^{-1} P^\top \log(I_\mathrm{e}) \\ \theta_\mathrm{e} &= c_1 \frac{a_1(1 - a_2 c_2)}{\sqrt{\sum \boldsymbol{I}_{\mathrm{e}i}}} \end{aligned}$$

where $\boldsymbol{w} = \begin{bmatrix} -10.5 & -9.5 & -8.5 & \dots & 9.5 & 10.5 \end{bmatrix}^\top$, c_1 and c_2 are scalars. The dashed line in Figure 9.16 shows the result of estimating the orientation of the edge using this method based on polynomial fit.

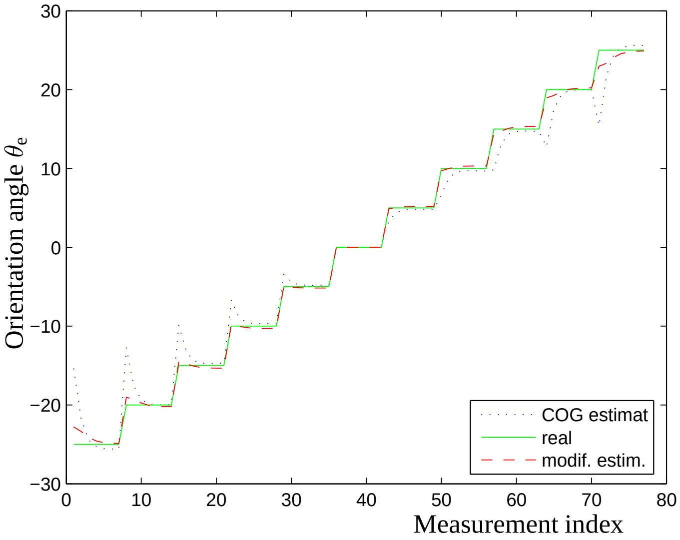


Figure 9.16: The result of the COG and polynomial fit orientation estimate for an edge.

From Figure 9.16 it is clear that the orientation estimate based on polynomial fit is better that the COG estimate. The estimate based on fitting a polynomial of degree two has a significantly smaller deviation from the actual orientation than the COG orientation estimate.

On both Figure 9.15 and 9.16 are the x axis index for the measurements and the y axis is the orientation angle. The simulated measurements have different distances from the center of the sensors, a total of seven different distances have been used. For each orientation of the object the distance from center of the sensors to object have been varied form 10cm to 70cm starting at 10cm in the indexing.

9.4 Summary

This chapter has described modifications to the 3D Reflection Map Model of Chapter 4 to model reflection from environment primitives such as a plane, an edge, a corner and a cylinder. The modified models has been validated in laboratory experiments. Also a method has been proposed to find the orientation of a plane and an edge based on reflection measurements. This method has only been tested with modeled reflection intensities.

Chapter 10

Conclusion and Future Work

The aim of this research project, was to develop a method for estimating 3D positions from measurements of intensity of reflected light, using an array of infrared emitters and receivers. This has been accomplished. Also accomplished are results for 2D position estimation, using a very simple algorithm, and some results on using the sensors for robot navigation sensors.

10.1 Conclusion

In Chapter 2 the theory of light and light reflection was introduced, and a number of different light reflection models was reviewed. Based on this review the Lambertian reflection law was selected as appropriate reflection model for this work.

Chapter 3 describes the sensors used in the laboratory experiments, together with the signal processing used to modulate and demodulate the emitted/received light using the Rudin-Shapiro transform. Also the measurement noise of the sensors was investigated, and was shown to be Gaussian.

The 3D Reflection Map Model developed in Chapter 4 is a geometrically based reflection model. The 3D Reflection Map Model models the intensity of light emitted by a point source, reflected by a spherical object, that reaches the receiver. The model incorporates the positions, orientations and characteristics of the light emitter respectively the light receiver, and the position, size and reflection properties of the spherical object. Laboratory experiments has been carried out to validate the 3D Reflection Map Model, and the model was found in good accordance with the measured intensities of reflected light.

Two methods was proposed in Chapter 5 for estimating position from reflection intensities. Both methods relay on the 3D Reflection Map Model and searches the parameter space for best fitting object position. The first method proposed is a “naive” search where a grid of possible positions is tested and refined until sufficient precision is reached. This method is very computationally demanding, because many evaluations of the 3D Reflection Map Model are needed in every iteration. The second proposed method for estimating the object position is based on the Nelder-Mead simplex algorithm to search the space of positions in the model. This search algorithm uses only a few evaluations of the 3D Reflection Map Model for each iteration step in the search process, an is therefore more efficient when considering computational load. Chapter 5 ends with showing that the proposed method based on the Nelder-Mead simplex algorithm gives good results when using modeled reflections.

Chapter 6 contains results of applying the Nelder-Mead simplex algorithm based method for position estimation from reflections to real measured reflection intensities. The method is shown successful for reflections of a ball object, and the robustness and computational demands of the method is investigated. This investigation results in choosing a subset of reflection measurements, for implementation in a non-Touch Screen prototype, based on the proposed method. The robustness of the method allow the ball object to be replaced by a human hand and still be functional. Finally the positioning of sensors are investigated with respect to optimal placement, and some results are found which indicate that the sensor setup used for the non-Touch Screen is not optimal.

During a three month visit to Alcalá University it was proposed to use the sensors for a human-machine interface (HMI) for a wheelchair. The HMI should make it possible for the wheelchair user to control the motion of the wheelchair by head movements. The method for such a HMI is described in Chapter 7. This work lead to the development of a simple method for determining the two dimensional position of an object directly from the measured reflection intensities. Furthermore a patent application has been filed for the position detection method, the patent application number is PA 2005 01217, and it is entitled "Method for touch-free human-machine interface".

The HMI for wheelchair control has been implemented on the SIAMO wheelchair of Alcalá University, as described in Chapter 8. The prototype implementation has shown that the HMI works as expected and that a wheelchair can be controlled precisely with this HMI.

In Chapter 9 reflection models of environment primitives has been developed, based on the 3D Reflection Map Model of Chapter 4. The environment primitives included are plane, edge, corner and cylinder. Laboratory measurements has been used in validating the models, and there is good accordance between measurements and model. The simple direct method for determining 2D position of an object has been adapted to use for estimating orientation of objects for use in a mobile robot. The orientation estimation method has been successfully tested on modeled reflections from a plane and an edge.

10.2 Future Work

The work on this project has resulted a number of useful methods, but there are still areas where additional work is needed.

The proposed methods for estimating 3D position from reflections in Chapter 5 is based on the 3D Reflection Map Model of Chapter 4. For this model to be accurate is needs to be calibrated mainly due to variations in emitter characteristics and object reflectivity. The calibration procedure used relay on reflection measurements of the object in a grid of positions in the region of interest. Development of a simpler calibration procedure is highly desirable.

The 3D position from reflections method proposed are only tested with a fixed set of reflection measurements. The encoding/decoding method used in modulating the light opens the possibility to estimate the noise on the measurements, as proposed in [16] and [19]. This could be utilized to only consider the reflection measurements with best signal to noise ratio, and this way lower the computational demands in estimating the position.

Further work is also needed on the optimal sensor placement for the non-Touch Screen, and in general. In Chapter 6 the sensor setup for the non-Touch Screen was found not to be optimal in the sense considered there. So further investigation and experiments are needed to validate the results obtained with respect to optimal sensor placement.

Considering the HMI for control of a wheelchair, further work is needed to find a suitable set of commands, especially the possibility for the used to switch the HMI on and off is desirable, to prevent unintended wheelchair movements e.g. when having dinner. Also the robustness against direct sunlight and other sources of light needs more work, to make sure the HMI is fully functional in all practical situations.

The results on navigation sensor for a mobile robots have not been tested experimentally, and the methods are not developed to the same extend as the other methods proposed in this thesis. To show that the proposed models and methods are useful in practice more work in this area is needed.

Appendix A

Material for Shielding

In parts of the project a positioning device have been utilized, for moving a ball to predefined positions, with the purpose of measuring the intensities of the infrared light reflected from this ball. Therefore the positioning device needed to be coated/shielded such that the reflection of infrared light from the positioning device was limited to a minimum, and there by not disturbing the measurements of light intensities reflected from the ball. Also the infrared LED's and photodiodes used has the unwanted property of emitting respectively being sensitive to light on/from the rear side, so shielding was also needed here to be able to fit both emitter (LED) and receiver (photodiode) into the same housing.

A.1 Non-Reflective Materials

In order to find a material that was good at *NOT* reflecting infrared light a number of different materials was tested for their reflectivity of infrared light. The materials is listed in Table A.1, both materials that was expected to have low reflectivity and some which was expected to have high reflectivity for the infrared light used was tested.

The materials 1–7 are different paints for the test pieces of wooden stick, as material 8 with diameter 10mm was painted with the different types of paint. The materials 9–15 was formed to a cylinder also of diameter 10mm.

The test for reflectivity was made by attaching the stick/cylinder to the positioning device, and then sliding the material under test along a line in front of an emitter-receiver pair perpendicular to the direction of the emitter-receiver pair, as shown in Figure A.1, the distance from stick/cylinder is 60mm.

No.	Name	Color	Gloss
1	Blackboard paint	dark green	mat
2	Sericol screen ink, Polyplast PY	black	mat
3	Maraplak MM073	black	mat
4	Alflex 50 + 90 mix	black	semi glossy
5	Maraplak MM64	light green	mat
6	Maraplak MM067	grass green	mat
7	Maraplak MM059	royal blue	mat
8	Wood	-	mat
9	Self-adhesive vinyl film	black	mat
10	Self-adhesive vinyl film	gold	glossy
11	Self-adhesive vinyl film	black	glossy
12	Self-adhesive vinyl film	medium green	glossy
13	Coated plastic	black	mat
14	Self-adhesive vinyl film	transparent	glossy
15	Card board	black	mat

Table A.1: Materials tested for reflectiveness of infrared light.

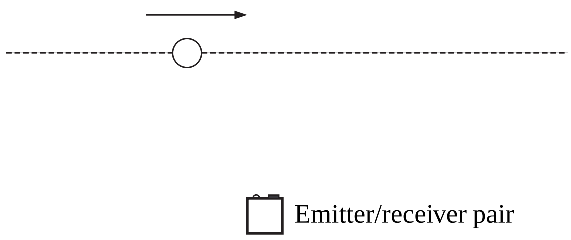


Figure A.1: Setup for measuring reflectivity of different materials.

The intensity measurements of the reflected infrared light for the different materials and a measurement with no stick/cylinder present are shown in Figure A.2 where the different sub-plots are numbered as in Table A.1. Note, that the y axis is different from sub-plot to sub-plot. The units are sample number on the x axis and sampled value on the y axis.

The sub-plots 12, 3, 4, 9, 11, 13 and 15 is obtained from measurements on black materials, all mat except materials 4 and 11 which are glossy, and the relative high fluctuations on sub-plot 4 are due to the fact that a wooden stick is not perfectly circular, but more like to a circle with an overlaid cosine, giving more peaks as there are several angles where specular reflection is high. For material 11 the spikes is assumed to have a similar origin, because the cylinder of self-adhesive vinyl film is not perfectly circular. The measurement marked “no” in the lower left hand sub-plot in Figure A.2 is a measurement

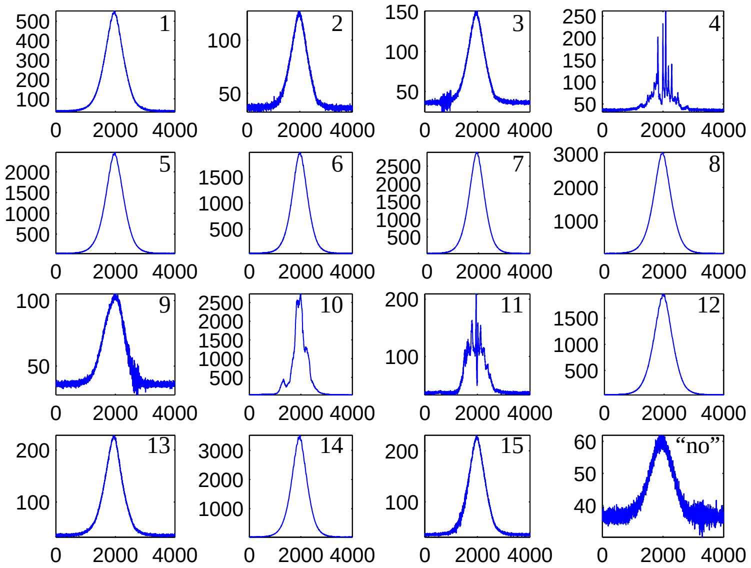


Figure A.2: Measurements of reflected infrared light from the different materials.

without a stick/cylinder present and shows the background reflection resulting from the positioning device. Subtracting this background "noise" from the measurements gives the result shown in Figure A.3, still with different y axis.

Based on the measured intensity of the reflected light shown in Figure A.3 the materials can be grouped into three groups, black materials, blackboard paint, and other materials. The green blackboard paint is the only one with measured intensities in the range 250–1500. The various black materials have all measured peak reflected light intensities less than 250 both for mat and glossy materials. The remaining materials tested have all measured peak reflected light intensities above 1500 with the darker green ones being in the lower end and the transparent self-adhesive vinyl film having the highest reflectivity.

As expected the black materials had the least reflectivity of infrared light. Amongst these there are quite some differences in reflectivity, the one with lowest reflectivity is material no. 4, mat black self-adhesive vinyl film which has a measured peak value of less than 45, and in the other end (not considering the the glossy ones) is no. 15, black card board with a measured peak value of about 165.

The measurements have been replicated with a distance of 120mm from emitter/receiver pair to stick/cylinder passing by. The result of these measurements are shown in Figure A.4. Compared to Figure A.3 the measured peak values for all materials are reduced with a factor of approximately 10. Also note, that the for the two glossy black materials

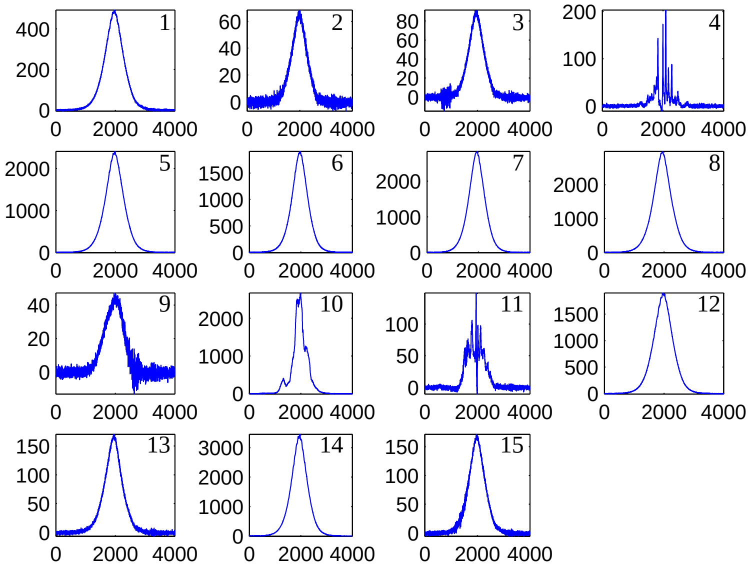


Figure A.3: Measurements of reflected infrared light from the different materials, corrected for background reflections, distance to sensor 60mm.

(no. 4 and 11) there are high spikes. This is because the distance from stick/cylinder to the measuring photodiode is increased, and the physical area of the light sensor results in a low-pass filter of the incoming light. Because the stick/cylinder is sliding by at this larger distance, the time where the specular reflection is present is very short and hence eliminated by the low-pass filtering effect.

The mat black self-adhesive vinyl film is the best of the materials in this test for shielding against unwanted reflection of infrared light.

It is worth noting, that black blackboard paint was not available at the time of this investigation is expected to reflect very little infrared light. Using this material for shielding was suggested to the author by Steen G. Hanson from Risø.

Other materials with low reflectivity for infrared light has been found during the work is black anti-static foam used for IC’s and the black plastic from anti-static bags. Both materials seems to have good properties when used in situations where reflections of infrared light is unwanted.

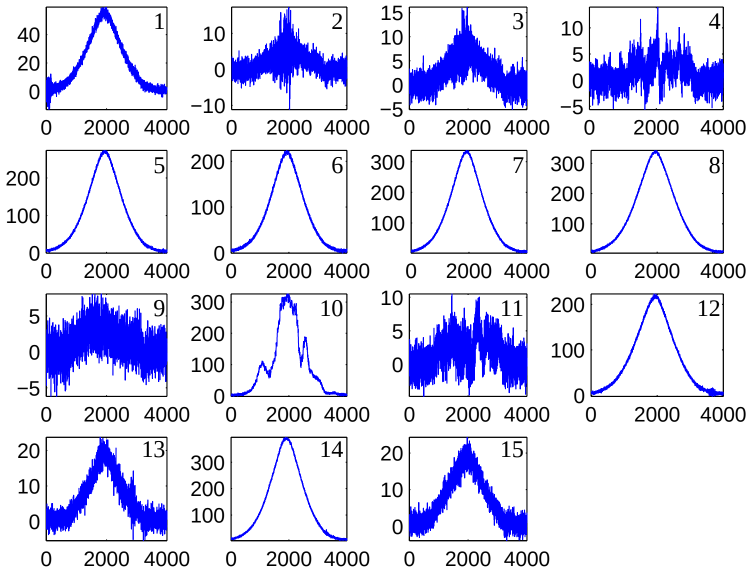


Figure A.4: Measurements of reflected infrared light from the different materials, corrected for background reflections, distance to sensor 120mm.

A.2 Blocking Infrared Light

An other type of shielding found necessary was to prevent infrared light from being emitted into the housing of the emitter/receiver pairs from the back end of the LED's. It is unappropriate to have the infrared light emitted into the housing because the photodiode in the receiver part is sensitive to infrared light from the rear side as well as from the front side. The result of this in the first sensor build was that the light from the back end of the LED saturated the photodiode in the same housing.

Here all kinds of materials that is non-transparent to light can be applied, and one simple solution would have been to separate emitter and receiver in separate housings of e.g. aluminum. This solution was not appropriate because it had been chosen to have the emitters and receivers placed close to each other in pairs. So some material that could be used to block the infrared light within the housing had to be found.

Special Non-transparent paint made with the purpose of preventing light from shining trough was found effective when applied to both rear end of the LED and the back side of the photodiode, in a sufficiently thick layer.

Other materials used for interrupting infrared light is thick card board from cardboard boxes, and the black self-adhesive vinyl film, the black anti-static foam for IC's and black plastic from anti-static bags also used to prevent reflections, as it often has been found useful that the material used to interrupt infrared light do not reflect infrared light either, because placed close to an emitter/receiver pair.

Appendix B

Position Device

One of the key goals in this project is to make a 3D input device for a computer, where the position of the object is found based on output from low cost optical sensors. To be able to validate the system, and for use in the investigation process it is crucial to know the actual position of the object. A positioning device is needed to be able to position the object at a known position.

The position device have to fulfill the following list of requirements:

- Position in three-dimensional space
- Cover positions within $80\text{cm} \times 60\text{cm} \times 60\text{cm}$
- Reposition within 0.3mm
- Computer controllable
- “Invisible” to infrared light
- Low noise (electro magnetic)

The choice to custom construction of the position device was made. Three linear drive units, Elmore ECO 60 RSR2, three Panasonic servo motors with servo drivers, MUDS5A5A1A, was fitted to a frame as shown in Figure B.1.

The size of the three linear drive units make the three-dimensional positioning cover the required area. An interface to the servo drives gives computer controllability, and a reference point search gives the required precision. The low electro magnetic noise requirement was fulfilled by using the option to turn off the servo drives. Finally the position device was made reasonable “invisible” to infrared light, by mechanical shielding of the problematic parts.

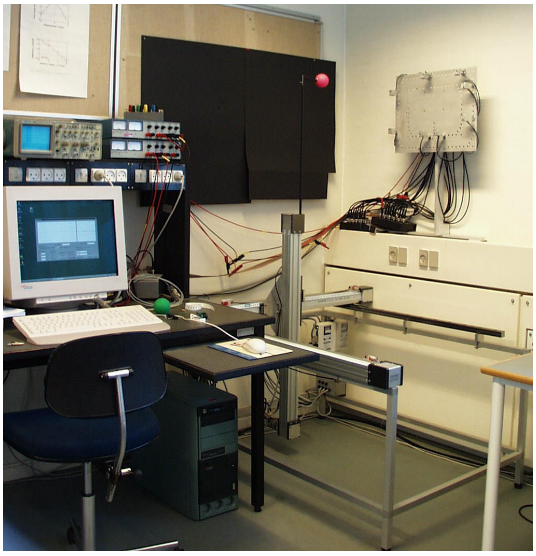


Figure B.1: The position device used to position the object in a known position.

Bibliography

[1] T. Aytac and B. Barshan. Differentiation and localization of target primitives using infrared sensors. In *Proc. of IEEE/RSJ Int. conference on Intelligent Robots and System*, volume 1, pages 105–110. IEEE, October 2002. ISBN: 0780373987.

[2] T. Aytac and B. Barshan. Rule-based target differentiation and position estimation based on infrared intensity measurements. *Optical Engineering*, 42(6):1766–1771, 2003. ISSN: 00913286.

[3] L.M. Bergasa, M. Mazo, A. Gardel, R. Berea, and L. Boquete. Commands generation by face movements applied to the guidance of a wheelchair for handicapped people. In *Proc. of 15'th International Conference on Pattern Recognition*, volume 4, pages 660–663, 2000.

[4] Nicolas Blanc, Thierry Oggier, Gabriel Gruener, Jan Weingarten, Alain Codourey, and Peter Seitz. Miniaturized smart cameras for 3d-imaging in real-time. In *Proc. of IEEE Sensors*, pages 471–474, October 2004. ISBN: 0780386922.

[5] H.V. Christensen. 3d reflection map modeling for optical emitter-receiver pairs. In *Proc. of IEEE Sensors*, pages 1241–1244, October 2004. ISBN: 0780386922.

[6] H.V. Christensen. Retrieval of 3d-position of a passive object using infrared led's and photodiodes. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 4, pages 1093–1096, March 2005.

[7] H.V. Christensen and J.C. García. Infrared non-contact head sensor, for control of wheelchair movements. In *Assistive Technology: From Virtuality to Reality*, volume 16 of *Assistive Technology Research Series*, pages 336–340. AAATE, IOS Press, September 2005. ISBN: 1-58603-543-6.

[8] D. Gareth Evans, Roger Drew, and Paul Blenkhorn. Controlling mouse pointer position using an infrared head-operated joystick. *IEEE Transactions on Rehabilitation Engineering*, 8(1):107–117, March 2000.

[9] J.C. García, H.V. Christensen, and M. Mazo. Interfaz hombre-máquina mediante un sensor de posición sin contacto por infrarrojos. In *Actas del XII Seminario Anual de Automática, Electrónica e Instrumentación Insdustrial (SAAEI 2005)*, pages 539–542. SAAEI, September 2005. ISBN: 84-8102-964-5.

[10] J.C. García, M. Mazo, L.M. Bergasa, J. Ureña, J.L. Lazaro, M. Escudero, M. Marron, and E. Sebastian. Human-machine interfaces and sensory systems for an autonomous wheelchair. In *Proc. of the AAATE'99 conference*, volume 6 of *Assistive Technology Research Series*, pages 272–277. AAATE, IOS Press, 1999.

[11] J.C. García, M. Mazo, J. Ureña, M. Marron, M. Escudero, and E. Sebastian. Modular architecture for wheelchairs with environment integration capabilities. In *Proc. of the AAATE'99 conference*, volume 6 of *Assistive Technology Research Series*, pages 214–219. IOS Press, 1999.

[12] C. Jennings. Robust finger tracking with multiple cameras. In *Proc. Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, pages 152–160, September 1999.

[13] A. Jensen and A. la Cour-Harbo. *Ripples in Mathematics, The Discrete Wavelet Transform*. Springer, 2001. ISBN 3-540-41662-5.

[14] C.T. Kelly. *Iterative Methods for Optimization*. Frontiersin Applied Mathematics. SIAM, 1999. ISBN 0-89871-433-8.

[15] Robert H. Kingston. *Optical Sources, Detectors, and Systems; Fundamentals and Applications*. Optics and Photonics. Academic Press, 1995. ISBN: 0-12-408655-1.

[16] A. la Cour-Harbo. *Robust and Low-Cost Active Sensors by means of Signal Processing Algorithms*. PhD thesis, Aalborg University, August 2002. ISBN 87-90664-13-2.

[17] A. la Cour-Harbo. The symmetric rudin-shapiro transform - an easy, stable, and fast construction of multiple orthogonal spread spectrum signals. In *Proc. of the 3rd Int. Symposium on Image and Signal Processing and Analysis (ISPA)*, volume 1, pages 262–265, September 2003. ISBN: 953184061x.

[18] A. la Cour-Harbo, A. Hernandez Alonso, H.V. Christensen, J.C. García García, M. Mazo Quintas, J. Stoustrup, and J. Ureña Ureña. Method for touch-free human-machine interface. Patent application, September 2005. PA 2005 01217.

[19] A. la Cour-Harbo and J. Stoustrup. Using spread spectrum transform for fast and robust simultaneous measurement in active sensors with multiple emitters. In *Proc. of IECON 02*, volume 4, pages 2669–2674, November 2002.

[20] Jeffrey C. Lagarias, James A. Reeds, Margaret H. Wright, and Paul E. Wright. Convergence properties of the nelder–mead simplex method in low dimensions. *SIAM Journal on Optimization*, 9(1):112–147, 1998. ISSN: 10526234.

[21] C.R. Lomba, R.T. Valadas, and A.M. de Oliveira Duarte. Experimental characterisation and modelling of the reflection of infrared signals on indoor surfaces. *IEE Proc. Optoelectron.*, 145 No.3:191–197, June 1998.

[22] M. Mazo. An integral system for assisted mobility. *IEEE Robotics & Automation Magazine*, 8(1):46–56, March 2001.

[23] J.A. Nelder and R. Mead. A simplex method for function minimization. *J. Computer*, 7(4):308–313, 1965.

[24] M.W. Nelisse. Integration strategies using a modular architecture for mobile robots in the rehabilitation field. *Journal of Intelligent and Robotic Systems*, 22(4):181–190, 1998.

[25] P.M. Novotny and N.J. Ferrier. Using infrared sensors and the phong illumination model to measure distances. In *Proc. of International Conference on Robotics and Automation*, volume 2, pages 1644–1649. IEEE, May 1999.

[26] B.T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975. ISSN: 00010782.

[27] J.M. Rheg and T. Kanade. Digiteyes: vision-based hand tracking for human-computer interaction. In *Proc. of the IEEE Workshop on Motion of Non-rigid and Articulated Objects*, pages 16 –22, November 1994.

[28] W. Rudin. Some theorems on fourier coefficients. In *Proc. of the American Mathematical Society*, volume 10, pages 855–859. American Mathematical Society, December 1959.

[29] P.B. Taylor and H.T. Nguyen. Performance of a head-movement interface for wheelchair control. In *Proc. of the 25th Annual Int. Conference of IEEE Engineering in Medicine and Biology Society*, volume 2, pages 1590–1593, September 2003.

[30] K.E. Torrance and E.M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57:1105–1114, September 1967.

[31] Mladen Victor Wickerhauser. *Adapted Wavelet Analysis from Theory to Software*. A K Peters, 1994. ISBN: 1-56881-041-5.

[32] Y. Sato, Y. Kobayashi, and H. Koike. Fast tracking of hands and fingertips in infrared images for augmented desk interface. In *Proc. of the Fourth Int. Conference on Automatic Face and Gesture Recognition*, pages 462–467, March 2000.