SCANNING AND MODELLING OF 3D OBJECTS

Jan Sternberg

LYNGBY 1997
IMM-EKS-1997-12

IMM

Trykt af IMM, DTU

ISSN 0909-6256

Acknowledgement

This project is the conclusion of my study at the Technical University of Denmark. The work has been conducted at the Department for Mathematical Modelling(IMM) in cooperation with Danisco.

It has been an inspiring experience to work with the image group at IMM, and all members of the staff as well as fellow students are thanked for their helpful attitudes and their ability to create a good atmosphere. A special thanks is directed to my supervisor Jens Michael Carstensen for supportive guidance.

The people at Danisco Development Center, Ib Frydendal, Lars Bo Jørgensen, and Frank Hald are thanked for being helpful on the establishment of this project, for delivering test objects (sugar beets), and for economical support for equipment investments.

Thanks are directed to the Institute of Surveying and Photogrammetry as well as Department of Manufacturing Engineering at the Technical University of Denmark too for the courtesy of letting me use their equipment.

Lyngby, February 15, 1997.

Jan Sternberg

Abstract

The work documented in this thesis is initiated by inspiration from the vision project at Danisco Development Center in Nakskov and the interest for 3D modelling at IMM.

Danisco is currently investigating the possibilities for the introduction of machine vision for the purpose of automatic classification, volume estimation, and top percentage estimation of sugar beets.

Ordinary 2D image analysis methodologies have been deployed, but for further enhancements the concept of 3D vision has been introduced. In the initial state of the project it has been of interest to obtain full scale 3D models of sugar beets for an investigation of geometric shape properties helpful for segmenting larger conglomerates of beets. A primary goal for investigating shape properties of sugar beets is also to estimate the location of the top slice.

This report deals with the construction of a laser scanner which is used to obtain 3D coordinate sets describing an object. The scanner is build using off-the-shelf components and is therefore constructed within a very limited budget.

The performance of the apparatus is investigated, and it is deployed in scanning a number of sugar beets. Data handling of the raw data is conducted for the formation of range images and to enable the usage of 3D visualization tools.

Finally some preliminary investigations on the shape properties of sugar beets are conducted.

Keywords: 3D modelling, sugar beets, shape properties, laser scanner, triangulation, camera calibration.

Contents

Chapter 1

Range Finding Techniques in Computer Vision

1.1 Introduction

Human vision is operating based on a number of factors from which three-dimensional information is deduced[10][6][7]. Sensory information usable for three-dimensional scene interpretation include occlusion effects (occluded objects appear farther away), texture gradients, apparent object size (magnification of closer objects), continuity of object outline (incomplete objects appear distant), shading, binocular perspective and stereo disparity, motion parallax (as we move around distant objects will not move across the retina as fast as close objects), scattering from the atmosphere (blue light is scattered in the atmosphere introducing an association between haziness and distance).

A great deal of the human depth interpretation is therefore based on experience and knowledge. On behalf of their nature the human methods are not all well suited choices for adoption in computer algorithms since they in a mathematical sense are ambiguous. Though it is possible to produce depth estimations using most of the above mentioned cues in computer vision, disparity of the binocular perspective or stereo vision is the geometrically most unambiguous method. This makes it a good candidate for exploitation in computer vision.

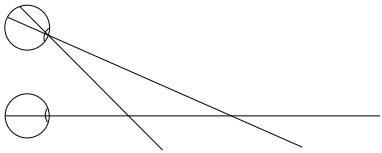


Figure 1.1: Principle of horizontal parallaxes in human vision

1.2 Computer Vision

A number of methods have been applied for non-contact metrology in computer vision and the subject is a continuously expanding field of research interest.

In general, range finding techniques have been categorized into two categories:

- Active range finding
- Passive range finding

Active techniques include methodologies such as ultra sound and light time-of-flight (e.g. radar) and triangulation systems utilizing an active component. They all involve transmission of some sort of energy and detection of its reflection.

Passive techniques generally only involve detection of energies/information already existing in the environment regarded and are thus purely image analysis oriented. Examples are shape from texture (texture gradient analysis), occlusion effects and depth from focusing. Also photogrammetry belongs to this category in its ordinary style using two cameras and no active component. In between the two categories methods using for example grey coded lighting are found.

1.2.1 Stereo Vision

For the human vision, the spatial scene perception of close range objects, is mostly performed by stereo vision. The basic concept is illustrated in figure 1.1.

The figure illustrates the concept of horizontal parallaxes: Points at different distances are projected on different parts of the retina. This difference

is referred to as the horizontal parallaxes and is used for range derivation performed by the brain. This concept is easily modeled by two cameras and a computer and is widely utilized in triangulation systems. The method constitutes the basis of the work in this thesis too, though one camera is exchanged with a laser.

1.2.2 Range Finding Techniques Involving Lasers

Lasers have been used in different approaches for the deduction of range information. Range may be calculated from the time space between the transmission and coaxially return of a laser pulse reflected by the object of interest. This method, however, needs very fast and precise electronics and the precision is often poor.

In other applications range is measured from the phase shift of a modulated continuous wave reflected by the object.

In this thesis a sheet of laser light is beamed onto the object and the reflection is observed with a camera. With knowledge of the relative orientation of the laser beam and the camera it is possible to calculate good range data from triangulation. The greatest benefit of this method compared to the two others is the cost. A well performing range scanning is in this manner constructed for a very limited cost. The main drawback of the latter approach is effects of having the sensor dislocated from the laser source leading to a number of problems described in section 2.5.1.

Chapter 2

Constructing the Laser Scanner

This chapter includes a description of the physical construction of the laser scanner and the software implementations for controlling the scanner. Finally the geometry of the system is described with an emphasis on range deduction and considerations on resolution and accuracy.

2.1 Background and Motivation

A number of considerations constitute the basis of the ideas for the construction of the laser scanner.

First of all, the objects for which a scanner is constructed set certain constraints on the dimensional properties of the scanner. In this case the measurement of sugar beets was the aim requiring the view of the scanner to be nothing less than 250mm.

Secondly, the total cost of the equipment needed to be kept within a limited budget. This encouraged the use of components already on hand and the employment of standard off-the-shelf equipment. As a result the final solution is easily disassembled and used in other applications and then later reassembled.

The versatility of the construction is of importance when constructing laboratory equipment: The laser scanner is not an industrial component needed to solve one and only one problem. Therefore it is appreciated that the scanner is up- and down-sizable.

Finally, the scanner of course has to meet demands of accuracy. The question of accuracy is in contrast to the conservation of a limited cost forcing the need of compromises.

Since time usage for the scan was not a critical factor, high speed implementation has not been a primary goal.

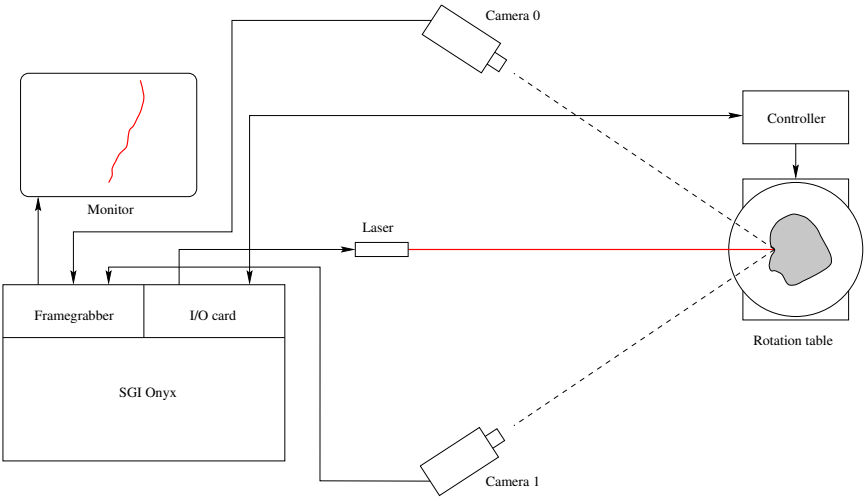


Figure 2.1: The laser scanner system.

2.2 The Scanner System

Figure 2.1 shows a schematic diagram of the entire scanner system.

The system consists of a number of main components: A rotary table including a planetary gear and a step motor, motor controller, two CCD-cameras, a laser, an SGI Onyx computer board with VME-bus, frame grabber, and an Acromag I/O-card. Besides the main components wiring and additional electronic components are used for interconnecting and controlling the equipment.

2.2.1 The Computer Processor

The Silicon Graphics Onyx (IP21) was chosen as the brain and heart of the system. This is to some extent overkill. However, the system was already equipped with an I/O-card and a well performing frame grabber. The frame grabber is a high performance Series 150/40 from Imaging Technology and it has a number of powerful features which may come in handy for a high-speed implementation. Those were not fully exploited in this

implementation. Also the system supports OpenGL and is equipped with Open Inventor, which was extensively used for visualization purposes.

2.2.2 The I/O-card

The I/O-card is a 9510-X VME multi-function card from Acromag. It is featuring both analog and digital I/O. Only digital ports are used for controlling the rotary table. Among features for the digital I/O-card are[2]:

- Four programmable 8-bit ports
- Read back on output states
- Output sinking up to 100mA from an up to 30V source
- Built-in protection diodes for driving relays directly
- Input range 0 to 30V
- TTL and CMOS compatible

For a description of the features implemented in the construction of the laser scanner consult section 2.2.6.

2.2.3 CCD cameras

The sensory system is composed of 2 Burle TC-354X CCD cameras with a resolution of 752x582 picture elements on a 1/2" CCD-array. Both cameras are equipped with 12mm Cosmicar/Pentax lenses featuring focus and aperture fixation. Lenses with the possibility of locking the settings are an advantage since loose rings may drift over time invalidating the calibration. As seen on figure 2.1 the cameras are placed on both sides of the laser plane to enhance the systems resistance to occlusion effects (see section 2.5.1). The cameras are tilted 90 degrees from their normal operation position to take better advantage of the image format scanning oblonged objects in vertical position.

2.2.4 The Laser

The laser is 10mW laser from Lasaris [1] with optics generating a sheet of laser light. The intensity in a crosssection of the sheet has a Gaussian distribution whereas the intensity profile along a line generated from a projection of the sheet upon a plane perpendicular to the beam direction is

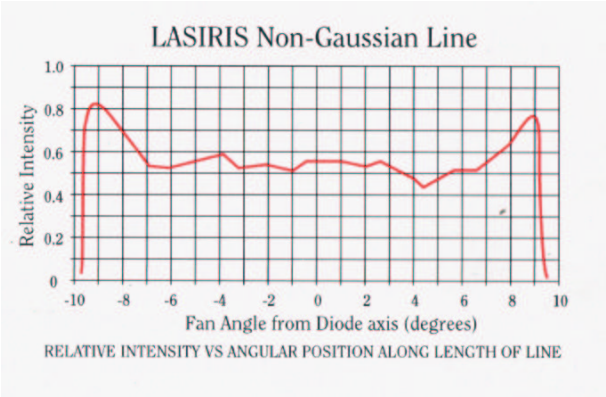


Figure 2.2: Laser intensity profile along laser line.

almost constant. The latter profile is illustrated on figure 2.2. Those properties of the laser beam are of great importance for the signal processing treated in section 5.1.

2.2.5 The Rotation Table and the Controller

All measurements performed by the laser scanner are done in the same plane. Therefore the surface scanned needs to be translated or rotated through the laser plane while successive measurements are done. As both front and back side of the object are required measured, rotation is the solution.

Precision rotary tables are available as preassembled complete entities. Those are highly prized and offer high accuracy. Since the accuracy required for this application is met at lower criteria, and the possibility for adding own features to the physical dimensions and properties of the rotation table was needed, the choice was to build the construction from scratch using selected components.

The only custom manufactured parts of the laser scanner are the platform, the rotation disk, and the calibration plate. The platform of the rotation table is constructed of 8mm steel plate making it a rigid and heavy platform resistant to vibrations and disturbances from the environment. Aluminum

profiles were attached to hold cameras and the laser. The profiles are of the multi-purpose type and thus enable reconfiguration of the system (e.g. using other lenses at a closer range or scanning at another scale) without any problems. Adjustment screws make it possible to bring the table into alignment.

The disk on which objects are placed for scanning is an aluminum disk of diameter 320mm. The disk is prepared for fixation of the calibration plate and customized object holders. For scanning sugar beets a special object holder was constructed as a three winged spike.

The rotation disk is mounted on a planetary gear which is driven directly by the step motor. The gear has a gearing of 10.5:1. The backlash is stated to be 15 arc minutes [17]. The usage of gears in an application like this always needs to be done with care since backlash is harmful to the accuracy. However, if the rotation is done in only one direction the influence of backlash is restricted. Doing rotations at a steady low speed without gaining too much inertia also limits the influence of backlash. Besides higher step resolution, one benefit from deploying gears is of course decreasing demand to the power of the motor driving the system.

The motor is the SM88.1.18.J3.248 4 phase step motor from Zebotronics [20]. The step motor has a resolution of 200 steps in 360 degrees, i.e. 1.8° per step and the positioning precision is 3% (3.2 arc minutes).

A step resolution of 1.8° is pretty coarse. The coarseness is decreased by the gear revealing a step resolution of 0.17° (appr.10 arc minutes). Another way to do even smaller steps is to electrically control the current in the motor coils and thus try to position the motor in between to physical steps. This feature is referenced to as ministepping.

The motor controller used for controlling the step motor is the SMD40B manufactured by JVL electronics [11]. The controller is featuring 10, 25, 50, and 125 ministep functionality, a possibility of setting the hold and drive current of the motor, overload protection, as well as galvanical isolation. It has inputs for step pulses, a direction signal, and an error signal is output in case of overload or malfunction.

A ministep configuration of 10 ministeps is used in this application. This gives a final count of 21000 steps on a 360 degrees rotation. The controller is in theory configurable to 125*200*10.5=262500 step per 360 degrees, but the ministeps need to be used with precaution. If the ministep functionality is to work perfectly, extremely high class motors with excellent linearity are

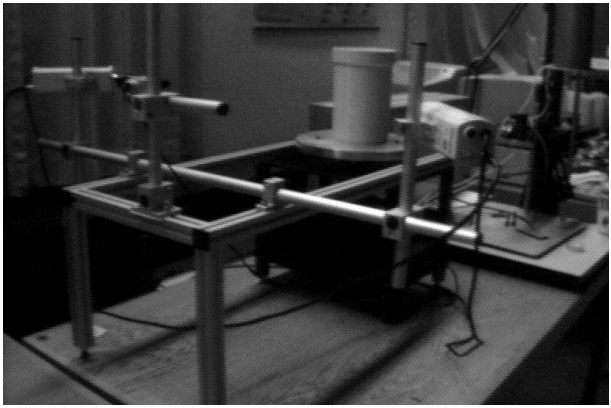


Figure 2.3: The laser scanner.

needed. Otherwise one step in the forward direction may in reality cause the motor to turn in the opposite direction due to the loss of perfection in the placement of the coils in the motor.

For the class of the SM87.1.18 motor 10 ministeps (the minimal configuration of the controller) is considered to fully exploit the capabilities of the motor. The controller works within a voltage range of 15-80V. A 80V200W power supply is used in this case.

An image of the system is seen in figure 2.3.

2.2.6 Connecting Units

Figure 2.4 shows the wiring and the electronic components used for interfacing the motor, the control unit, the laser, and the I/O-card.

The digital port 3 on the I/O-card is used to control the scanner. The following points are connected:

- Pin 1 + 49: 5V power supply.
- Pin 46 + 48: Ground
- Pin 3: Protect circuit hardwired to 5V
- Pin 31: Point 0: Error signal in
- Pin 29: Point 1: Relay control out (power on/off)

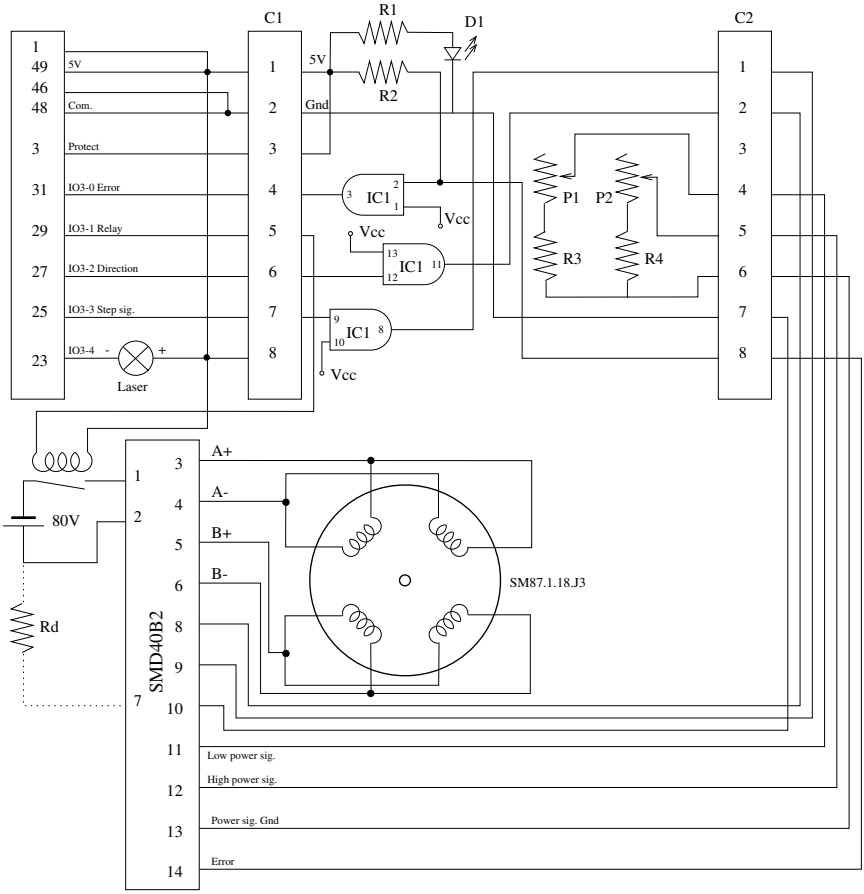


Figure 2.4: Wiring diagram for the control system.

- Pin 27: Point 2: Direction signal out
- Pin 25: Point 3: Step signal out
- Pin 23: Point 4: Laser control out

As stated above the I/O-card is able to sink 100mA. This is a sufficient current to drive the laser directly from the I/O-card since the laser only uses 70mA at 5V. It is also enough power to drive the relay coil for switching the power on and off. Main power control possibility is useful to have in case an error signal is received and the system needs to be turned of to prevent further damage.

The other outputs, the step and direction signal, are buffered using an AND port. The step motor controller needs 12-30 mA on the step and direction signals. This is too much load for the open collector circuit of the I/O-card if a well performing system is expected. Since the signal is of both polarities the relatively high sink current of the I/O-card does not solve the problem. One solution is to change the pull-up resistors of the card, but a set of buffers come in more convenient and also work as a cheap protection of the I/O-card in case of extreme and hazardous voltages accidently should be short cut to the circuit. A HC7408 circuit was chosen. It has got relatively high output rates and is flexible to voltage ranges. Also the input of the error signal is buffered with an AND gate.

The extra inputs on the AND gates may be utilized in future expansion for further control of the rotation table. E.g. for installation of extra safety by controlling the error signal from a secondary source.

The resistor network constituted by R_3, R_4, P_1, and P_2 determines the hold and drive current of the motor. The currents are adjustable within the ranges 0.5A to 6A respectively 3A to 6A.

2.3 Controlling the System

The laser scanner is entirely controlled through the I/O-card. The laser and the rotary table are turned on and off, the direction and step signals are written directly to the port, and the controller error signal is sampled.

The read/write operations to the I/O-card are performed as DMA-operations. The process of controlling system entities in this way becomes a matter of setting bits in the appropriate address space(Port3: base + 0x9d).

The main concern on controlling the rotation table is concentrated on the generation of the step signal where considerations on timing need to be done.

2.3.1 The Generation of a Step Clock Signal

With the controller set to 10 ministeps, the 10.5:1 gearing and 200 steps for a full 360 degrees rotation of the step motor, 21000 pulses need to be clocked on the step signal output to rotate the rotation table one revolution. If the rotary table is deployed in high speed applications this is quite a significant number of pulses to generate. At normal operation, neither the I/O-card nor the controller are subject to bottleneck problems. The motor controller is operating at frequencies up to 800kHz.

The main problem is the software implementation of signal generation. The issue is subdivided into two issues: Problems regarding processor time-sharing on a multi-user system and problems on getting high frequency precision timer signals for clocking the step signal.

Timers

The Irix-operation system supports the BSD Unix feature of timers known as itimers. An itimer is capable of sending a signal at the expiration of a requested time interval. The timers work as one-shot or repetitive timers. Of the different kinds of itimers the itimer of type ITIMER_REAL, which measure elapsed time, is of interest concerning real-time programming. However, itimers are not able to deliver high frequency timer signals. Even when running a non-degrading process using the *fasthz* tuning parameter the highest timer frequency achieved is no higher than 2400Hz. For the step signal generation two timer signals are needed for every step signal (one to signal the positive edge of the pulse and one signaling the negative edge). All in all the maximum frequency of the step signal is 1200Hz corresponding to 17.5s for a complete revolution of the rotation table.

The solution of the problem is to use the Irix cycle counter. The cycle counter is part of all Silicon Graphics systems. It is a free-running counter updated by hardware at high frequency. For Onyx systems (IP21 boards) the counter is a 64-bit integer counter which is incremented every 21000 nanoseconds, i.e. the update frequency is 47.6MHz.

It is possible to map the image of the counter into process address space, where the value can be sampled. Doing repetitive readings of the counter value it is then possible to generate pulse signals whenever the counter has incremented a specific value. The high update frequency of the counter prevents the same value to be read at successive read-outs since the CPU instruction rate is not much higher. While the process is not interrupted the CPU instruction rate becomes the limiting factor of the step signal frequency. At normal operation in a multi-user environment a 30kHz signal was easily achieved.

Time-sharing systems

Running real time processes on systems where the CPU(s) are shared among a number of processes and/or users does often involve difficulties. Using the cycle counter for the generation of high frequency signals as described above, the signal will suffer drop-outs due to interrupts of the CPU from other processes and the kernels scheduling interrupts. In normal operation the kernel pauses the CPU every 10 milliseconds to make scheduling decisions. This time interval is referred to as one tick and a process is normally executing in a guaranteed time slice of 3 ticks. The CPU is scheduling processes according to their priority level.

The avoidance of the interrupts in the execution of the process is achieved utilizing different features: High priority non-degrading process execution, prolonged time slice duration, isolation of a CPU from scheduled work and finally turning off the dispatching "tick" for the CPU making the CPU non-preemptive. This will make a real-time process running as real-time as possible. Programming real-time on an Irix platform is most conveniently done with the React/Pro Frame Scheduler library. The library, however, was not available, and the real-time requirements of the laser scanning are limited as long as time usage is not considered the main issue.

The main requirements set by the laser scanner originates from the limitations on accelerating the motor. Due to the nature of the construction of step motors, the motor needs to gain speed before further acceleration is done. If the step signal is increased from zero to for example 1000 full steps per second instantly, the signal transition will be too fast for the motor to react (the motor lacks time for rotating from one position to the next before the hold-signal is received). The result is that the motor is not stepping at all but is only producing a buzzing sound. If program execution is done in

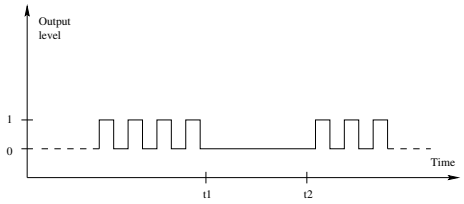


Figure 2.5: Typical step signal for loaded systems. t_1: Process with higher priority is scheduled for execution. t_2: Step signal process regains CPU resources.

normal mode, the step signal can not be guaranteed from drop-outs. The drop-outs will instantly stop the motor which has to be accelerated again and a high frequency step signal will be devastating for the position control and in worst case for the motor as well. The timing of the step signal is illustrated in figure 2.5.

If no other user is consuming CPU resources the time interval $t_2 - t_1$ is negligible at low frequency signals. If the CPU is heavily loaded the length of the time interval is increased and the rotation of the rotary table is subject to jittering. As long as the buzz-effect at too high acceleration rates is avoided the position of the rotation table is, within the mechanical accuracies, completely known relative to the start position. The interruptions in the time interval have no influence on the positional information in this case.

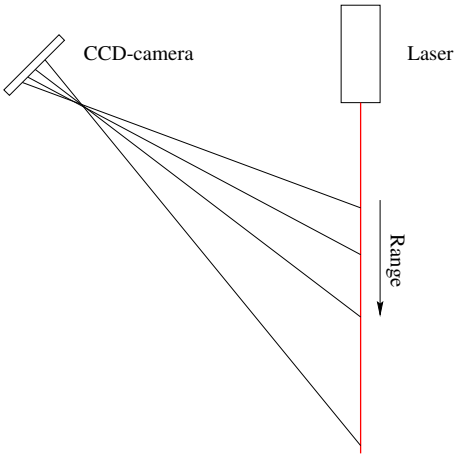


Figure 2.6: Basic principle of the range finding technique.

2.4 Geometry of Laser Scanner

This section describes the geometry of the laser scanner as well as some of the factors dictating the resolution of the scanner. Some familiarity with optical terms (see e.g. section 3.2.1) is assumed.

Basic Principles

The basic principle of range finding techniques using triangulation from a sheet of laser light and a CCD-camera is illustrated on figure 2.6. This is the machine implementation of figure 1.1. The laser sheet is perpendicular to the paper.

As illustrated, any range measure has got a unique corresponding offset in the CCD-array. Figure 2.7 shows the same setup from a view perpendicular to the laser sheet.

The CCD-camera is oriented with the rows of the CCD-array close to parallel to the laser plane and the range information is then seen as an offset in the direction of the row index. The first data processing done is the transform from the 2D image information of the CCD-readout to a set of

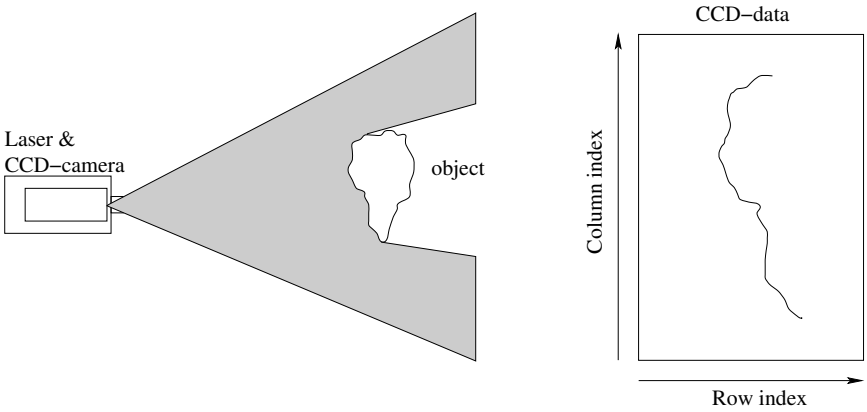


Figure 2.7: Range finding principle and sensor offsets.

offset values. The set of offset values are found scanning every column locating the laser beam location measured in rows. The location is preferably estimated with subpixel accuracy using an algorithm with a reliable performance. Section 5.1.1 gives an explanation on the signal processing. In the actual implementation the image information was sampled in a 512x768 format in the framegrabber resulting into 768 offset values, of which some may be marked invalid if no laser light is found. To generate a complete range image either the object or the scanner system is moved to another known position and the process is repeated. The offset values need to be transformed into real world 3D-coordinates. This is done directly if the orientation of the laser plane relative to the calibrated CCD-camera is known, or it may be done indirectly by calibrating the CCD-camera to the laser plane calculating a transform from image coordinates to laser plane coordinates.

The latter approach was preferred for this application. The calibration is performed by orienting a calibration plane to be coincident with the laser plane. The calibration of the camera(s) thus results in a transform from camera coordinates to image plane coordinates. Three dimensional coordinates are then calculated from the known movement (rotation) of the object. Consult chapter 3 for a thorough treatment of the calibration theory.

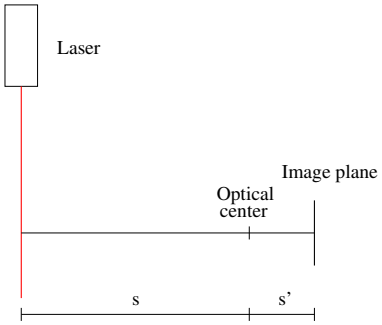


Figure 2.8: The entire laser plane in focus.

Focus

To get a well focussed image on the image plane, the relations dictated by the lens law must be obeyed[12]:

$$\frac{1}{s} + \frac{1}{s'} = \frac{1}{f}$$

where f is the focal length, s is the distance to the object and s' is the optical image distance where the theoretical sharpest image is formed. s and s' are the orthogonal distances from the optical center. In ordinary CCD-cameras, the image plane is normally mounted perpendicular to the optical axis and s' is constant over the entire sensor area. If focus for the entire laser plane is wanted, the lens law has to be satisfied and thus the optical axis needs to be perpendicular to the laser plane too as depicted in figure 2.8.

The advantage of a well focussed image in this setup is in most applications outnumbered by the disadvantages. If a planar object which is parallel to the optical axis is scanned no laser light will reach the sensor if the object is above the optical axis. This drawback is to some degree avoided if the object scanned has only little geometrical variation. In this case the setup is tilted 45 degrees such that surface normals of the object bisect the perpendicular angle (α) between the optical axis and the laser sheet.

In the implemented laser scanner the view angle is approximately 45 degrees and the image is thus accepted to be out of focus in some areas. The

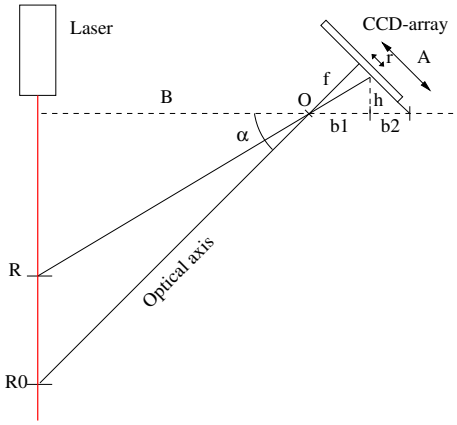


Figure 2.9: Geometry for range derivation.

influence of noncompliance to the lens law is to some degree minimized by increasing the depth of focus by using a small aperture. Also the usage of moment based algorithms for localizing the laser light makes the influence of out-of-focus areas less significant, since the blur will be close to symmetrical, assuming Gaussian optics [8].

Range Equations

As mentioned, range information is calculated from a plane-to-plane transform. However, the derivation of the directly calculated range does reveal some general properties on the resolution of the system and some good qualitative understanding of design issues.

With reference to figure 2.9 the range is seen to be:

$$R = \frac{Bh}{b_1} \tag{2.1}$$

B is also referenced as the baseline which is the distance between the optical centers of the laser and the camera lens. f is the camera constant of the camera lens, r is the sensor offset position, α is the viewing angle, the angle between the optical axis of the camera and the baseline, and R is the

range. It is noted that the range could just as well be defined relative to the optical center of the camera lens as to the laser lens. It does however make more sense to use the laser plane as reference since, in this application, coordinate information is measured in the laser plane.

From simple geometric considerations the following equations are derived:

$$b_1 + b_2 = \frac{f}{\cos\alpha}$$

$$A = (b_1 + b_2)\sin\alpha = f\tan\alpha$$

$$h = (A - r)\cos\alpha$$

$$b_2 = (A - r)\sin\alpha$$

Bringing it all together with (2.1), the basic range equation becomes:

$$\begin{aligned} R &= B\frac{(f\tan\alpha - r)\cos\alpha}{f(\frac{1}{\cos\alpha} - \frac{\sin^2\alpha}{\cos\alpha}) + r\sin\alpha} \\ &= B\frac{f\tan\alpha - r}{f + r\tan\alpha} \end{aligned} \tag{2.2}$$

As the offset position is discretized into sensor elements with an equidistant spacing, the non-linear relationship of (2.2) results in varying range precision at constant sensor precision. The derivative of (2.2) is

$$\begin{aligned} \frac{\delta R}{\delta r} &= B\frac{-(f + r\tan\alpha) - \tan\alpha(f\tan\alpha - r)}{(f + r\tan\alpha)^2} \\ &= -B\frac{f(1 + \tan^2\alpha)}{(f + r\tan\alpha)^2} \\ &= -B\frac{f}{cos^2\alpha(f + r\tan\alpha)^2} \\ &= -\frac{Bf}{(f\cos\alpha + r\sin\alpha)^2} \end{aligned} \tag{2.3}$$

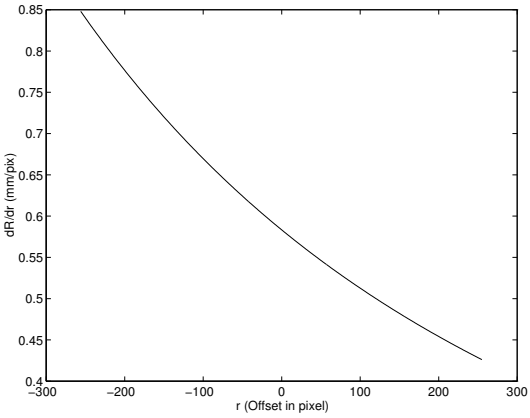


Figure 2.10: Varying precision of range values at different offsets. $f = 12mm, \alpha = 45\,\text{deg}$, pixels pacing: $8\ \mu m, B = 0.35m$.

This relationship is plotted in figure 2.10 using parameters close to the true values of the configured system. $r = 0$ corresponds to the r-value of the principal point (the intersection of the optical axis and the image plane).

At normal placement of the object on the rotation table, measurements are typically done at offset values $r > 0$ resulting in a resolution of approximately 0.5mm/pixel.

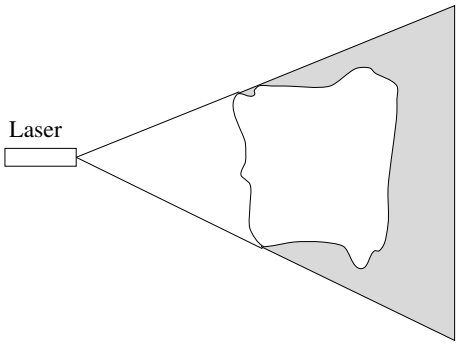


Figure 2.11: Laser occlusion.

2.5 Error Sources

The laser scanner is vulnerable to a number of error sources. First of all the geometry of the setup has build-in limitations to the shape of the objects scanned, but also the inhomogeneity of the reflectance and surface normals of the objects add errors to the measurements. The errors of the latter type are treated in section 5.1.2 since they are inherent to the assumptions on which the signal processing rely.

2.5.1 Occlusion Effects

When using methods of triangulation the problems of occlusion arise. Occlusions are divided into two categories: Laser occlusions and camera occlusions. Laser occlusions appear whenever there are parts of the object which the laser light can not reach. The phenomenon is illustrated in figure 2.11

The shaded areas are not reachable for the laser light and coordinate information from those areas is therefore not obtainable (unless a rotation of the object later will enable laser light to reach the designated point). One way to avoid laser occlusion is to move the laser further away from the scanned object. Moving the laser further away, nevertheless, will increase the width of the laser beam due to focusing limitations. Since the sensors are subject to similar occlusion, there is no need to move the laser much further away from the object than the cameras (distances measured in the laser plane).

Another kind of occlusion is sensor occlusion. Sensor occlusion appears whenever the sensing camera is restricted from seeing the laser light due to the objects self occluding effect. Especially objects with an irregular surface, sharp edges and cavities are subject to sensor occlusion. Figure 2.12 shows a case of camera occlusion. The shaded area is not visible from the position of camera 0. A reduction of the triangulation angle (i.e. moving the camera closer to the laser) is one way to minimize the probability of the phenomenon to occur. Decreasing the angle however increase the uncertainty of the coordinate calculation. At zero degrees the coordinate triangulation totally fails.

A better solution is to use more than one camera for sensing the laser light. Constructing the laser scanner it was chosen to place an extra camera on the other side of the laser plane. As illustrated in figure 2.12, this solution reduce the likelihood for a total occlusion though deep holes still may cause a double camera occlusion.

Adding another camera to the system may also improve coordinate estimates from the additional redundancy.

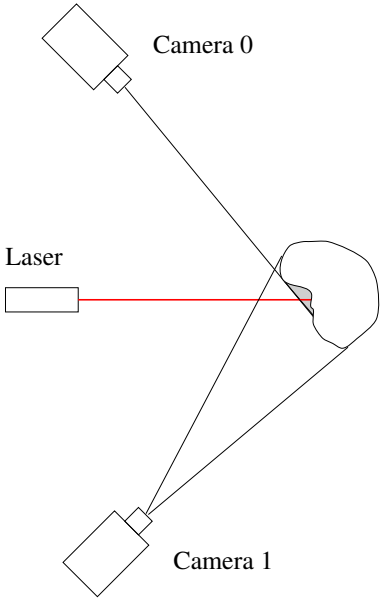


Figure 2.12: Limiting camera occlusions.

Chapter 3

Camera Calibration

For the purpose of deriving coordinates in the laser plane the cameras need to be calibrated to the laser plane. In this chapter two calibration methods are described.

3.1 Introduction

Camera calibration is the process resulting in a set of estimates of the parameters describing the transformation from 3D world coordinates to 2D camera coordinate. The parameters estimated from camera calibration are the intrinsic parameters which describe the interior geometry and optical characteristics of the camera, and the extrinsic parameters describing the position and orientation (rotation) of the camera relative to the world coordinate system.

The method for the calibration and the number of parameters estimated in the process may indeed depend upon the application and its need for an exact description of the world coordinate to image coordinate transformation.

Parameters commonly estimated are:

Intrinsic Parameters

- f - The focal length of the lens
- $\kappa_1, \kappa_2 ...$ - (radial) Lens distortion
- (C_x, C_y) - The intersection of the optical axis and the sensor element(CCD frame)
- β, s_x - Affine deformation/scaling due to errors in CCD-chip to frame grabber link

Extrinsic Parameters

- ψ, θ, ϕ - Rotation of camera coordinate system relative to world coordinate system
- T_x, T_y, T_z - Translation of camera coordinate system relative to world coordinate system.

The purposes for a camera calibration may be divided into two classes.

- Inference of 3D information from 2D image coordinates.
- Inference of 2D image coordinates from 3D information.

The inference of 2D image coordinates from 3D object coordinates is most commonly used for object inspection in assembly applications. This is accomplished as a hypothesis test where the location of objects in the world coordinate system is confirmed or declined from the appearance or disappearance in the expected 2D image coordinate system. Both extrinsic and intrinsic parameters need to be calibrated.

The inference of 3D information from 2D image coordinates is obtainable from different methods. One of the most well known applications is photogrammetry, where 3D information is deduced from 2D coordinates in two cameras or from two perspectives. With a calibrated camera a set of 2D image coordinates determines a ray in 3D space on which the object point imaged must lie. Using two cameras/images, homologous points - the images of the same object point in the two different images - will constitute 2 rays in 3D space. The 3D world coordinates of the object point are obtained as the intersection of those rays. For this purpose both the intrinsic as well as extrinsic camera parameters are needed. Therefore a full calibration must be conducted.

Photogrammetry is conducted at any scale. Geographical mapping is done from images acquired at high altitude(aircraft or satellite) and small-scale photogrammetry is performed for object inspection of e.g. circuit boards.

In fact photogrammetry can be used for almost any kind of metric measurement. The above mentioned applications all concern the measurement of objects or the determination of the location of a target in a 3D world coordinate system. Another kind of 3D information deduced from a camera view is the location and orientation of a moving camera relative to the world coordinate system (e.g. for robot vehicle guidance).

In fact 3D object coordinates may be derived from only one image if other constraints of the object are known. If the object lies in a certain known plane in 3D space, the object point is determined from the intersection of the plane and the ray constructed from the image point and the camera parameters. This fact is exploited in this paper. For the determination of 3D object points, a laser plane is projected onto the object. As the plane parameters are known, the intersection of the ray and the plane is obtainable from simple mathematics. This of course is only true if the line is not parallel to the plane.

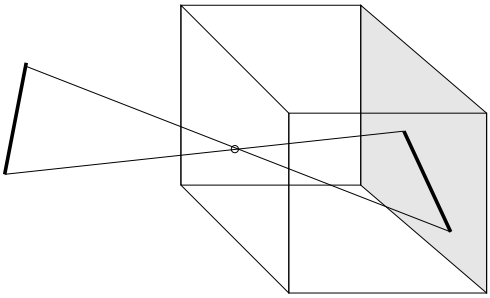


Figure 3.1: The pinhole camera model.

3.2 System Geometry and Coordinate Definition

3.2.1 The Perspective Projection and The Pin-hole Camera Model

Modelling the mathematics of cameras, the pinhole camera model is commonly used. This is due to the fact that the imaging of a camera is close to the central perspective projection described by the pinhole camera model. In fact some of the very first cameras were constructed as pinhole cameras. One of the big disadvantages of those cameras are the long times of exposition. When using optical lenses it is possible to focus larger bundles of light rays reducing the time of exposition drastically. The introduction of optics does on the other hand introduce some imperfections but the pinhole camera model is still performing well as a theoretical camera model. The model is illustrated in fig. 3.1.

In theory all rays pass undistorted through the well defined projection center O - also known as the optical center. The rays are imaged on the image plane which in this case is a CCD array. The line passing through the center of projection perpendicular to the image plane is called the optical axis. The distance f from the image plane to the center of projection is the focal length or camera constant.

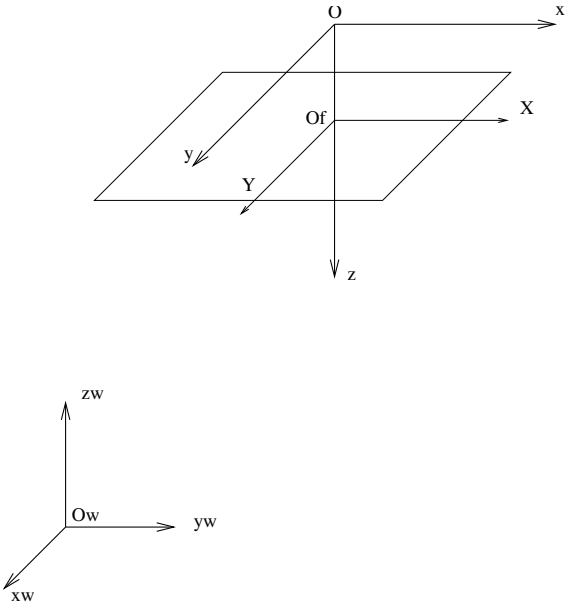


Figure 3.2: Model coordinate systems.

3.2.2 Coordinate Systems

Four coordinate systems are introduced. The three are illustrated on fig. 3.2.

Those are the world coordinate system (x_w, y_w, z_w), the camera coordinate system (x,y,z), and the image coordinate system (X,Y). A last coordinate system (X_f, Y_f) is introduced for the discrete array of pixels representing the image in the frame buffer. Coordinate systems and notation in this paper is mainly adopted from [18][19][13]. As illustrated front projection (image plane in front of projection center) is used.

3.2.3 Coordinate Transformation

The transformation from world coordinates (x_w, y_w, z_w) to camera coordinates (x,y,z) is performed as a 3D rotation of the world coordinate system

around the origin followed by a 3D translation to the origin of the camera coordinate system.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{T} \tag{3.1}$$

R is the 3 x 3 3D rotation matrix:

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \tag{3.2}$$

And **T** is the translation vector

$$\mathbf{T} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{3.3}$$

The 3D rotation is separable into three independent rotations about the three coordinate axes. The order of rotation is significant to the content of the final rotation matrix. In most literature the primary(first) rotation takes place about the x-axis. Then a rotation is performed around the new y-axis followed by a final rotation around the z-axis. In this paper the order is adopted from [18] and [19] which is the opposite. First the primary rotation ψ of the $x_w y_w z_w$-system about the z-axis. From fig. 3.3 the relation for the transformation of $x_w y_w z_w$-coordinates of a point into the higher order $x_\psi y_\psi z_\psi$-system is recognized to be:

$$\begin{aligned} \mathbf{X}_\psi &= \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} \\ &= \mathbf{R}_\psi \mathbf{X_w} \end{aligned} \tag{3.4}$$

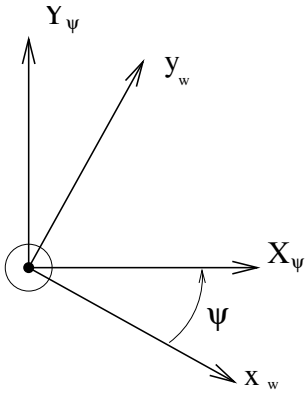


Figure 3.3: Primary rotation.

The secondary rotation θ of the $\mathbf{X}\psi$-system about the y_ψ-axis is from similar considerations seen to be described by the matrix:

$$\begin{aligned} \mathbf{X}_{\theta\psi} &= \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} x_\psi \\ y_\psi \\ z_\psi \end{bmatrix} \\ &= \mathbf{R}_\theta \mathbf{x}_\psi \end{aligned} \tag{3.5}$$

And the last rotation ϕ of the $\mathbf{X}_{\theta\psi}$-system about the $x_{\theta\psi}$-axis is defined by:

$$\begin{aligned} \mathbf{X}_{\phi\theta\psi} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_{\theta\psi} \\ y_{\theta\psi} \\ z_{\theta\psi} \end{bmatrix} \\ &= \mathbf{R}_\phi \mathbf{x}_{\theta\psi} \end{aligned} \tag{3.6}$$

It is noted that angles of rotation are defined as positive when rotating clockwise viewed from the origin of the coordinate system towards $+\infty$ along the axis of rotation.

Bringing it all (3.6), (3.5) and (3.4) together reveals the complete 3D rotation matrix

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_\phi \mathbf{R}_\theta \mathbf{R}_\psi \\ &= \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \end{aligned} \tag{3.7}$$

$$\begin{aligned} r_1 &= \cos\theta\cos\psi \\ r_2 &= \cos\theta\sin\psi \\ r_3 &= -\sin\theta \\ r_4 &= -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi \\ r_5 &= \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi \\ r_6 &= \sin\phi\cos\theta \\ r_7 &= \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ r_8 &= -\sin\theta\cos\psi + \cos\phi\sin\theta\sin\psi \\ r_9 &= \cos\phi\cos\theta \end{aligned}$$

Extensive explanation on 3D rotation of coordinate systems is found in [12][9][7][3]. It is noted that $\mathbf{R}$ is orthonormal.

To fulfill the coordinate transformation a translation of the rotated world coordinate system to the camera coordinate system is the final step. The translation is defined by the vector between the two coordinate systems origins:

$$\mathbf{T} = \overrightarrow{OO_w} \tag{3.8}$$

using camera coordinates.

In much literature the translation is performed before the rotation. This paper treats the coordinate transformation vice versa. The choice of order may be done arbitrarily, but it is noted that R is unique to the choice of order of translation/rotation as well as the order of the independent three rotations ϕ, θ, and ψ. In this case the rotation is followed by the translation, as described above. This procedure is adopted from [19] and is crucial for the proof of the Tsai-algorithm.

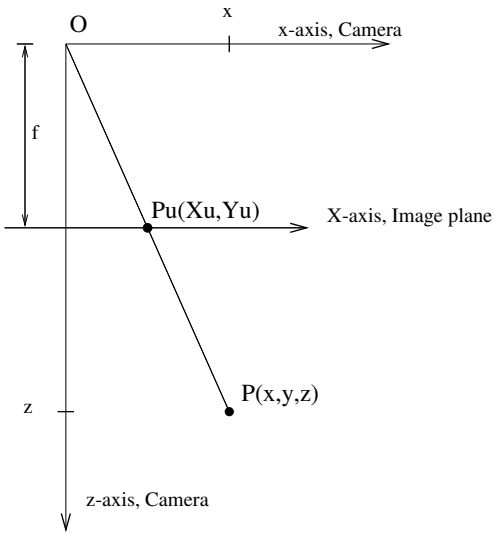


Figure 3.4: The perspective projection.

3.3 Inner Orientation

3.3.1 Central Projection

As stated in the description of the pinhole model, the 2D imaging of the 3D world takes place as a projective (or central) projection. The projection is illustrated in figure 3.1. Figure 3.4 illustrates the perspective projection of only one coordinate to simplify calculations.

From simple geometry inspection of figure 3.4 reveals the correspondence between camera x-coordinates and image plane X_u-coordinates:

$$X_u = f\frac{x}{z} \tag{3.9}$$

Similar for the y-component:

$$Y_u = f\frac{y}{z} \tag{3.10}$$

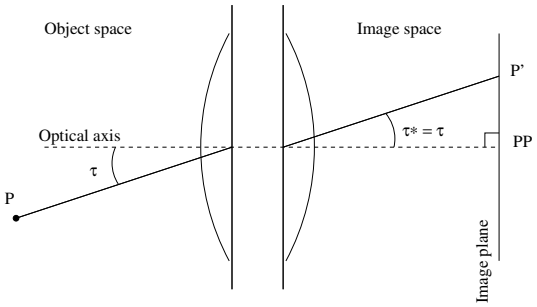


Figure 3.5: Optical system of image formation.

The coordinates (X_u, Y_u) are the undistorted image coordinates which in the ideal case (no lens distortion) will be what is imaged on the image plane.

3.3.2 Lens Distortion

However, due to lens distortions the ideal undistorted image coordinates will rarely be what is actually imaged. The fact that lenses are used instead of a pinhole adds imperfection to the image. Figure 3.5 illustrates the ideal path of a light ray passing the lens system.

Distortions introduced from imperfections of the lens system are often divided into radial and tangential lens distortion [14][9] [7][12][19][3]. If the ray enters the lens system at another angle as it leaves the lens system, there is radial lens distortion. If the incoming ray is not in the same plane as the ray leaving the lens, there is tangential distortion as well.

Due to production methods the radial lens distortion is the most pronounced of the two [9][19][3], and therefore most calibration routines only include radial lens distortion in the model. In the discussion of lens distortion polar coordinates are used. The optical axis is used as the origin of the coordinate system, and the coordinate set using polar coordinates (r, v) is the ideal location of the imaged point whereas (r_d, v_d) is the location in the distorted image. Figure 3.6 illustrates the definitions and results of lens distortions. D_r is the error introduced from radial lens distortion whereas D_t is the error due to tangential distortion.

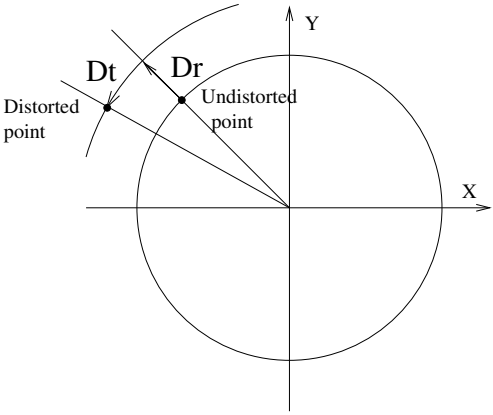


Figure 3.6: Definition of radial and tangential lens distortion.

As the tangential distortion is negligible, only the radial distortion is modeled:

$$r_d = r + D_r \tag{3.11}$$

r_d is the distorted radius and D_r is the radial distortion modeled as [9][18][19][7]:

$$D_r = \kappa_1 r^3 + \kappa_2 r^5 + \kappa_3 r^7 + \kappa_4 r^9 + ... \tag{3.12}$$

The projection on the coordinate axis reveals:

$$\begin{aligned} X_d &= X_u(1 + \frac{D_r}{r}) \\ &= X_u(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \kappa_4 r^8 + ...) \end{aligned} \tag{3.13}$$

$$\begin{aligned} Y_d &= Y_u(1 + \frac{D_r}{r}) \\ &= Y_u(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \kappa_4 r^8 + ...) \end{aligned} \tag{3.14}$$

$$r = \sqrt{X_u^2 + Y_u^2} \tag{3.15}$$

The radial lens distortion is often modeled as:

$$D_r = \kappa_1 r^1 + \kappa_2 r^3 + \kappa_3 r^5 + \kappa_4 r^7 + ... \tag{3.16}$$

instead of (3.12). This modelling is more correct but in a situation of calibration it only adds redundancy. From (3.9), (3.10), and (3.15) it is seen that

$$r = \frac{f}{z}\sqrt{x^2 + y^2} \tag{3.17}$$

$$\Downarrow$$

$$\frac{D_r}{df} = \frac{1}{z}\sqrt{x^2 + y^2} = \frac{r}{f} \tag{3.18}$$

In other words; a change of the focal length *df* implies a linear radial distortion with the magnitude:

$$D_r = r\frac{df}{f} \tag{3.19}$$

which means that κ_1 in (3.16) is a hundred percent correlated with changes of the focal length. Therefore this coefficient is included in the estimate of the focal length.

Above the radial distortion is a function of the distance from the principal point of the image (the intersection of the optical axis and the image plane) to the undistorted point. This is a natural choice for modelling, though there is no fundamental difference in choosing the distortion to be measured as a function of the distorted position. This is how the problem is treated in [18],[19], and [13].

(3.13) and (3.14) then become

$$\begin{aligned} X_u &= X_d(1 + \frac{D_r}{r}) \\ &= X_d(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \kappa_4 r^8 + ...) \end{aligned} \tag{3.20}$$

$$\begin{aligned} Y_u &= Y_d(1 + \frac{D_r}{r}) \\ &= Y_d(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \kappa_4 r^8 + ...) \end{aligned} \tag{3.21}$$

Radial lens distortion is often observed as barrel distortion (D_r negative using definition (3.13) and (3.14)) and pincushion distortion (D_r positive same definition).

3.3.3 From Undistorted Image Coordinates to Computer Image Coordinates

Left is the transformation of the real image coordinates (X_d, Y_d) to computer image (rows and columns in frame buffer) coordinates (X_f, Y_f):

$$X_f = s_x \frac{X_d}{d_x'} + C_x \tag{3.22}$$

$$Y_f = \frac{X_d}{d_y} + C_y \tag{3.23}$$

$$d_x' = d_x \frac{N_{cx}}{N_{fx}} \tag{3.24}$$

where d_x and d_y are the center distances between adjacent pixels in the CCD-camera in the x- and y-direction. s_x is an uncertainty scale factor introduced due to a variety of factors such as hardware timing mismatches.

C_x and C_y represent the frame coordinates for the intersection of the optical axis and the image plane. N_{cx} is the number of sensor elements in the x direction and N_{fx} is the number of pixels in a line sampled by the computer. d_x, d_y, N_{cx} and N_{fx} are usually known parameters supplied by the manufacturer.

The entire process of the transformation from 3D world coordinates to 2D computer coordinates is summarized in:

$$\frac{d'_x}{s_x}X + \frac{d'_x}{s_x}X\kappa_1 r^2 = f\frac{r_1 x_w + r_2 y_w + r_3 z_w + T_x}{r_7 x_w + r_8 y_w + r_9 z_w + T_z} \tag{3.25}$$

$$d_y Y + d_y Y\kappa_1 r^2 = f\frac{r_4 x_w + r_5 y_w + r_6 z_w + T_x}{r_7 x_w + r_8 y_w + r_9 z_w + T_z} \tag{3.26}$$

$$r = \sqrt{({s_x}^{-1} {d_x}' X)^2 + (d_y Y)^2}$$

3.4 Calibration

Two methods for camera calibration are described below:

- The direct linear transformation
- The Tsai two stage technique

The direct linear transformation is very simple and does only include a few parameters whereas the Tsai algorithm is a lot more sophisticated and includes more parameters.

3.4.1 The Direct Linear Transformation

If (non-linear) lens distortion is neglected the transformation from world coordinates to images coordinates is described solely by the transformation of a rigid body (3.1) and the perspective projection (3.9) and (3.10) where X_u and Y_u are considered being the true image coordinates. In frame coordinates (3.22) (3.23) the total transformation is given by:

$$\begin{aligned} {s_x}^{-1} d'_x X &= s_x^{-1} d'_x (C_x - X_f) \\ &= f \frac{r_1 x_w + r_2 y_w + r_3 z_w + T_x}{r_7 x_w + r_8 y_w + r_9 z_w + T_z} \qquad (3.27) \\ d_y Y &= d_y (C_y - Y_f) \\ &= f \frac{r_4 x_w + r_5 y_w + r_6 z_w + T_x}{r_7 x_w + r_8 y_w + r_9 z_w + T_z} \qquad (3.28) \end{aligned}$$

Reformulating (3.28) the establishment of two linear equations of twelve unknown parameters appears. Formulated in homogeneous coordinates [7] [15] the linear system is:

$$\begin{bmatrix} wX_f \\ wY_f \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \qquad (3.29)$$

Restricting one of the unknowns by e.g. setting a_{34}=1, the system is solved from 6 or more points using a least squares approach.

In the use for the laser scanner where the transformation is a plane to plane transformation (laser plane to image plane) the z-coordinate is constant and the transformation matrix is limited to a 3x3 matrix:

$$\begin{bmatrix} wX_f \\ wY_f \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \tag{3.30}$$

The direct linear transformation is convenient to use and is applicable in a great deal of situations where a limited accuracy is needed. It has got the advantages that no a priori knowledge about the inner orientation (besides that lens distortion is considered insignificant) is needed. Also, the use of linear calculus is without need of start guesses for the parameters. However, it has got a number of drawbacks too. The parameters are not physical camera parameters, though those are obtainable from further calculations. The physical properties of the rotation matrix are ignored and only its orthogonality will only appear as a special case. The method is also reported have some degree of instability [7].

3.4.2 The Tsai Calibration Technique

Introduction

In this section the calibration algorithm of Roger Y. Tsai is investigated. The algorithm is often referred to as Tsai's two-stage calibration technique. This section will describe the background and highlight the main features of the theory. Finally the actual algorithm is described. A complete description is found in [18],[19] and [13]. The implementation by Reg Willson was adopted and deployed with minor modifications to serve the scanner system.

The Tsai algorithm has been chosen among others for a number of reasons.

- The algorithm models lens distortion. When using off-the-shelf equipment lens distortion is significant and should therefore be included in the calibration, if a minimum of accuracy is required.
- Versatility.
- Coplanar calibration objects may be employed. Coplanar calibration objects are easier to produce than non-coplanar objects and are well constructed for laser-plane alignment.
- No start guess for calibrated parameters is needed. A minimum of user interaction is needed.

The version deployed calibrates estimates of the rotation matrix $\mathbf{R}$, the translation of the coordinate system $\mathbf{T}$, the focal length f, the radial lens distortion κ_1, and the scale factor s_x as well as the location of the principal point (C_x, C_y). The pixel spacing d_x and d_y, and the number of cells N_{cx} N_{fx} are parameters known a priori. The coordinates of the principal point are quite often assumed to be at the center of the image, but the true location may deviate a significant number of pixels from the assumed location. Better results may therefore be obtained if the optimization is performed for those parameters too.

Camera Model and Coordinate Systems

The camera model and the definitions of the coordinate systems of the two-stage technique is the same as described in section 3.2. There is one difference though in modelling the lens distortion. The Tsai technique

models the radial lens distortion as a function of the distorted image coordinates(radius). In this implementation only one parameter (besides the linear part included in the estimation of the focal length) κ_1 is estimated. Thus the distortion is defined as:

$$\begin{aligned} X_u &= X_d(1 + \frac{D_r}{r}) \\ &= X_d(1 + \kappa_1 r^2) \end{aligned} \tag{3.31}$$

$$\begin{aligned} Y_u &= Y_d(1 + \frac{D_r}{r}) \\ &= Y_d(1 + \kappa_1 r^2) \end{aligned} \tag{3.32}$$

$$r = \sqrt{X_d^2 + Y_d^2} \tag{3.33}$$

Motivation for the Tsai two-stage Technique

The two-stage technique is based on a number of facts. First of all the radial alignment constraint. From figure 3.7 it is observed that no matter the magnitude of radial lens distortion (no tangential distortion), the direction of vector $\overline{O_i P_d}$ is unchanged and radially aligned with $\overline{P_{oz} P}$. This observation is referred to as the radial alignment constraint and is a cardinal point in Tsai's camera calibration algorithm. For a complete proof for the two-stage algorithm consult [18] and [19].

The algorithm

The two stages of the algorithm are:

- Stage 1: Compute 3D orientation, x and y translation.
- Stage 2: Compute effective focal length f, lens distortion coefficient κ_1, and z translation.

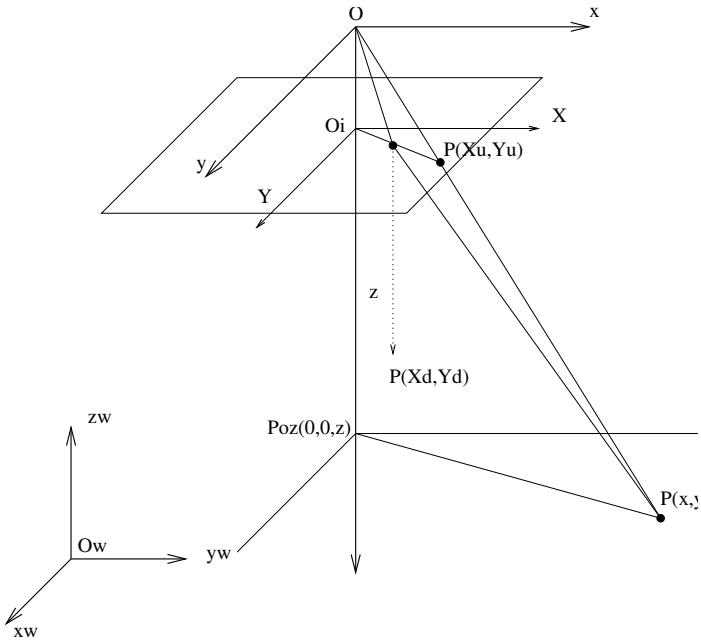


Figure 3.7: Radial alignment constraint.

Stage 1

In stage 1 five extrinsic parameters are estimated: The three rotations ϕ, θ, and ψ as well as the x and y translations T_x and T_y. The procedure is as follows.

Grab a frame into computer memory and calculate the distorted image points of the calibration object. Match image points with 3D coordinates of the calibration object. For every point i with coordinates (x_{wi}, y_{wi}, z_{yi}) and (X_{di}, Y_{di}) set up the linear equation:

$$\begin{bmatrix} Y_{di}x_{wi} & Y_{di}y_{wi} & Y_{di} & -X_{di}x_{wi} & -X_{di}y_{wi} \end{bmatrix} \begin{bmatrix} {T_y}^{-1}r_1 \\ {T_y}^{-1}r_2 \\ {T_y}^{-1}T_x \\ {T_y}^{-1}r_4 \\ {T_y}^{-1}r_5 \end{bmatrix} = X_{di} \tag{3.34}$$

With the number of points much larger than five, an overdetermined system is established for the solution of the unknowns ${T_y}^{-1}r_1$, ${T_y}^{-1}r_2$, ${T_y}^{-1}T_x$, ${T_y}^{-1}r_4$, and ${T_y}^{-1}r_5$. Equation (3.34) simply follows from the radial alignment constraint $\overline{P_{oz}P} \| \overline{O_iP_d}$. See [19] for detail. Next let $\mathbf{C}$ be a scaled 2×2 submatrix of $\mathbf{R}$:

$$\mathbf{C} \equiv \begin{bmatrix} {r_1}' & {r_2}' \\ {r_4}' & {r_5}' \end{bmatrix} \equiv \begin{bmatrix} {T_y}^{-1}r_1 & {T_y}^{-1}r_2 \\ {T_y}^{-1}r_4 & {T_y}^{-1}r_5 \end{bmatrix} \tag{3.35}$$

If $\mathbf{C}$ has full rank, compute

$${T_y}^2 = \frac{S_r - \sqrt{{S_r}^2 - 4({r_1}'{r_5}' - {r_4}'{r_2}')^2}}{2({r_1}'{r_5}' - {r_4}'{r_2}')^2} \tag{3.36}$$

$$S_r = {r_1}'^2 + {r_2}'^2 + {r_4}'^2 + {r_2}'^2$$

If $\mathbf{C}$ does not have full rank, compute

$$T_y{}^2 = \frac{1}{r_i{}'^2 + r_j{}'^2} \tag{3.37}$$

The elements in **C** are uniquely calculated from (3.34). **C** is the upper left 2×2 submatrix of the rotation matrix **R** scaled by $T_y{}^{-1}$. It is proved [19] that no 3×3 orthogonal matrices exist that differ in their 2×2 submatrices with a scale factor different from ± 1. Vice versa given a 2×2 submatrix of a 3×3 orthogonal matrix, the scale factor is given except the sign. For the derivation of the calculation of the scale factor in (3.36) and (3.37) consult [19].

The sign of T_y is determined from the following algorithm:

1. Pick an object point i with image coordinates (X_{di}, Y_{di}) away from the image center (C_x, C_y).
2. Pick a sign for T_y, say $T_y > 0$.
3. Calculate r_1, r_2, r_4, r_5, and T_x from the solution of 3.34. Calculate (x, y) from $x_i = r_1 x_{wi} + r_2 y_{wi} + T_x$ and $y_i = r_4 x_{wi} + r_5 y_{wi} + T_y$, $z_{wi} = 0$.
4. If the sign of x and X is the same and the sign of y and Y is the same, then the sign of T_y is correct ($\text{sign}(T_y) = +1$). Otherwise reverse sign ($\text{sign}(T_y) = -1$).

The validity of the algorithm stated above is proven from observations of the radial alignment constraint in figure 3.7: X_d must have the same sign as x and Y_d the same sign as y since (X_{di}, Y_{di}) and (x, y) are found in the same quadrant of their respective coordinate system. As z and f are positive this also follows from (3.9) and (3.10).

The rotation matrix **R**

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \tag{3.38}$$

is now calculated. The first four elements are determined from the derivations above. The orthonormality properties of the rotation matrix imply the following equations:

$$|r_3| = \sqrt{1 - r_1{}^2 - r_2{}^2} \tag{3.39}$$

$$|r_6| = \sqrt{1 - {r_4}^2 - {r_5}^2} \tag{3.40}$$

The orthonormality also implies, that the inner product of the two first rows in $\mathbf{R}$ must be zero. Thus, let s_3 and s_6 denote the sign of r_3 and r_6 respectively, leading to

$$\begin{aligned} r_1 r_4 + r_2 r_5 + s_3 |r_3| s_6 |r_6| &= 0 \\ &\Downarrow \\ s_3 s_6 &= -sgn(r_1 r_4 + r_2 r_5) \end{aligned} \tag{3.41}$$

As a first guess s_3 is chosen as $+1$, leaving

$$s_6 = -sgn(r_1 r_4 + r_2 r_5)$$

Using the orthonormality and right handed property of $\mathbf{R}$ for the last instance, the last three elements of $\mathbf{R}$ are determined as the outer product of the two first rows.

$$r_7 = r_2 r_6 - r_5 r_3 \tag{3.42}$$
$$r_8 = r_4 r_3 - r_1 r_6 \tag{3.43}$$
$$r_9 = r_1 r_5 - r_4 r_2 \tag{3.44}$$

where r_3 and r_6 are determined from (3.39) and (3.40) and the choice of sign stated above.

Now, the arbitrary choice of signs is leading to only a guess of $\mathbf{R}$.

For the validation of $\mathbf{R}$, an estimate of f is computed using (3.45). If $f < 0$, which is a model contradiction, the choice of signs must be reversed. Thus the signs of r_3, r_6, r_7, and r_8 are reversed.

Stage 2 - Focal length f, distortion coefficient κ_1, and z-translation T_z

Ignoring lens distortion use (3.26) for every calibration point to establish the following linear equations with f and T_z as unknowns (z_w=0):

$$f(r_4 x_{wi} + r_5 y_{wi} + r_6) - d_y Y_i T_z = (r_7 x_{wi} + r_8 y_{wi} + r_9) d_y Y_i \qquad (3.45)$$

This is equivalent to the direct linear transformation (3.28) and therefore the solution is only an approximation to a better estimate.

The values found for f and T_z are used as start guesses for the exact solution of f, T_z and κ_1 using a non-linear iteration algorithm on (3.26). The initial guess for κ_1 may be set to zero.

For a complete proof of the Tsai algorithm consult [18] [19] and [13].

3.5 The Calibration Plane & Feature Localization

3.5.1 Mapping Image Points to World Coordinates

Performing a camera calibration it is of great importance to have accurate measurements of image features with known world coordinates. The measurement of those features need to be of subpixel accuracy due to the relatively coarse resolution of the CCD-array. Using a 12mm lens and a sensor element spacing of e.g. 10μ, the object space resolution at 0.60m is 0.5mm (see e.g. figure 2.10).

Besides the basic need of accuracy a number of properties for the measurement method are appreciated. A robust and versatile method is of course preferred to a method which is subject to fluctuations of great magnitude generated from e.g. image noise or small physical deviations in the equipment setup.

Constructing the laser scanner, centroid estimation from center of gravity was used. The choice was mainly due to the robustness of the method and the possibilities for creating calibration objects using an ordinary laser printer.

3.5.2 Moment Based Subpixel Algorithms

Derivation type processes are often employed in a great deal of feature localization and detection techniques in image analysis. Moment based methods, however, are entirely based on integration(summation) processes. This fact may be a significant reason for the popularity of moment based algorithms, since in many cases summations are more easily handled than algorithms involving calculations of derivation type features.

The moment transform of an image function, f(x,y), is given by

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)dxdy, \quad p,q = 0,1,2...\infty \tag{3.46}$$

In the discrete case of an M×N image f(i,j) the integration becomes a double summation:

$$m_{pq} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} i^p j^q f(i,j), \quad p,q = 0,1,2...\infty \tag{3.47}$$

Moments are used for grey tone images as well as binary images. For instance moment based estimates reveal good results when deployed in laser stripe center localization in grey tone images. In the case of segmented binary images (3.47) degrades to

$$m_{pq} = \sum_{i,j} \sum_{\in \Omega} i^p j^q \tag{3.48}$$

where Ω denotes the region with foreground intensity (usually '1'). The moment m_{00} then denotes the area of the region or object Ω. The centroid estimate in the discrete case thus becomes:

$$\hat{c} = (\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}) = (\frac{\sum_\Omega i}{\sum_\Omega 1}, \frac{\sum_\Omega j}{\sum_\Omega 1}) \tag{3.49}$$

This estimate is often referred to as the center of gravity.

3.5.3 Preprocessing

If centroids are wanted for a number of individual objects and not the entire image some form of preprocessing is needed for segmentation of the image. I.e. the image is segmented into a number of regions $\Omega_0, \Omega_1, \Omega_2....$, $\Omega_i \cap \Omega_j = \emptyset, i \neq j$.

3.5.4 Centroid Estimation of a Disk

In continuous space the centroid of a circle is estimated without errors (3.47) if censor and object planes are coplanar. This still is the case if the

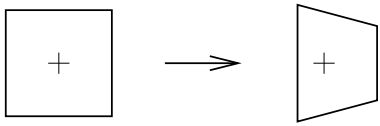


Figure 3.8: Perspective projection of a square.

planes are parallel and the projection is orthographic (and even the perspective projection if the projection center is on a line perpendicular to the center of the disk in object space). However, as soon as the sensor plane is tilted relatively to the object plane (or the projection center is moved relatively to the disk) a systematic error is introduced to the centroid estimation due to the properties of the perspective projection. From (3.9) and (3.10) it is clear that the perspective projection scales objects differently according to their distance from the projection center. The effect on centroid estimation is easily illustrated in the case of the perspective projection of a square. With the optical axis pointing at the center of the square and the sensor plane tilted (rotated around one axis only) there will be an error on one of the coordinates when calculating the centroid estimate. The visual effects are illustrated on figure 3.8. The crosses mark the estimated centroids - the shift of the centroid position of the perspective projected square is exaggerated to state the point.

The error is calculated from simple geometry and an integral.

The error on estimating the center of a square viewed from a gazing angle of 45° at the distance 450mm using a 12mm lens is plotted in figure 3.9. It is seen that regarding a square with side lengths 20mm the error is as much as almost 0.14mm corresponding to almost 0.4 pixels in image space with pixels spacing 10μ

Similar calculations were performed numerically for a disk of diameter 20mm and an object distance of 400mm using a 12mm lens. On figure (3.10) the error on the centroid estimated is shown as a function of the triangulation angle.

As it is seen above the systematic error of the centroid estimate originating from the effects of the perspective projection becomes ever more pronounced as the size of the object is increased. This fact is a plausible argument for keeping the calibration object as small as possible. On the other hand, small objects introduce new problems. First of all the moment

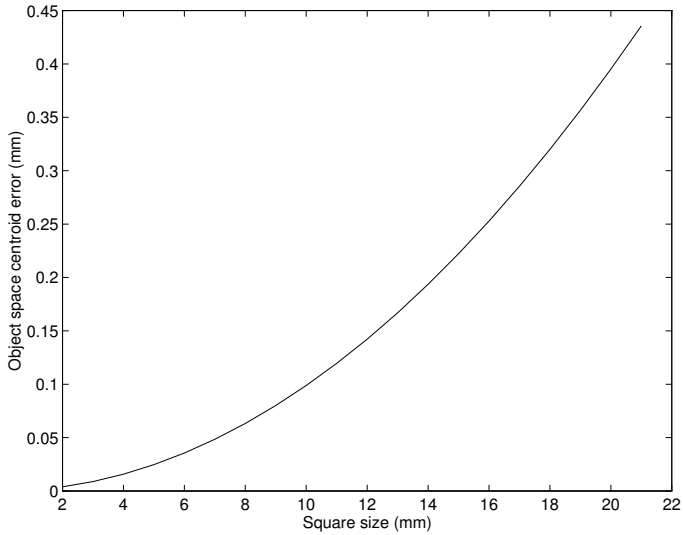


Figure 3.9: Errors on centroid estimation of a perspective projected square.

based centroid estimation of smaller objects is more vulnerable to noise. Also, the discrete version of the method introduces quantification errors. The smaller the object, the more significant is the error due to quantification. Figure 3.11 shows the from quantification originating error on the centroid estimate of a disk as a function of the disk radius. The data is numerically generated by letting the center of the disk take a large number of evenly distributed positions within one pixel and then from each position calculate the estimation of center and compare the result with the actual location.

3.5.5 Implementation

The Calibration Object

For the purpose of camera calibration a calibration object was created. The calibration object consists of an array of black doughnut shaped objects printed on a sheet of drawing film using an HP 4-L laser printer. The film was used in order to achieve resistance to fluctuating temperatures

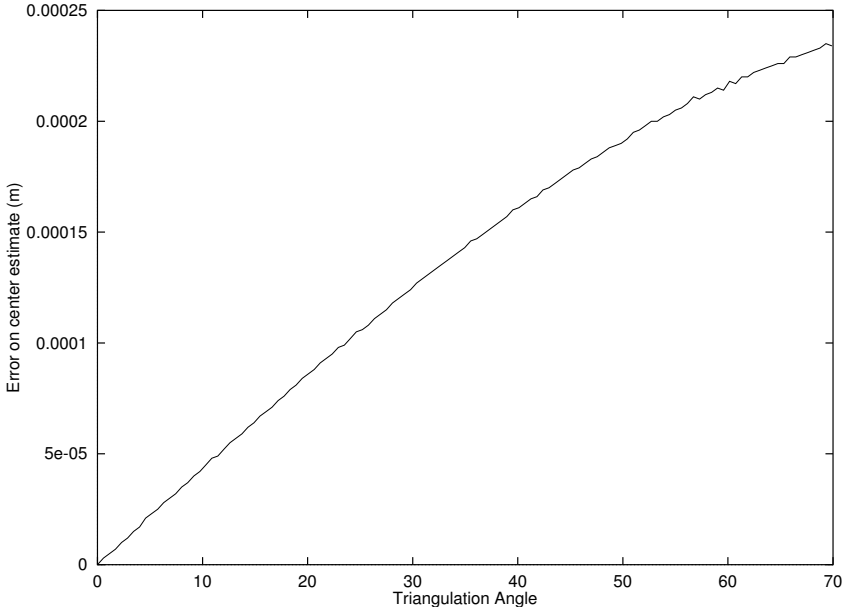


Figure 3.10: Errors on centroid estimation of a perspective projected disk at different angles. Disk diameter: 20mm

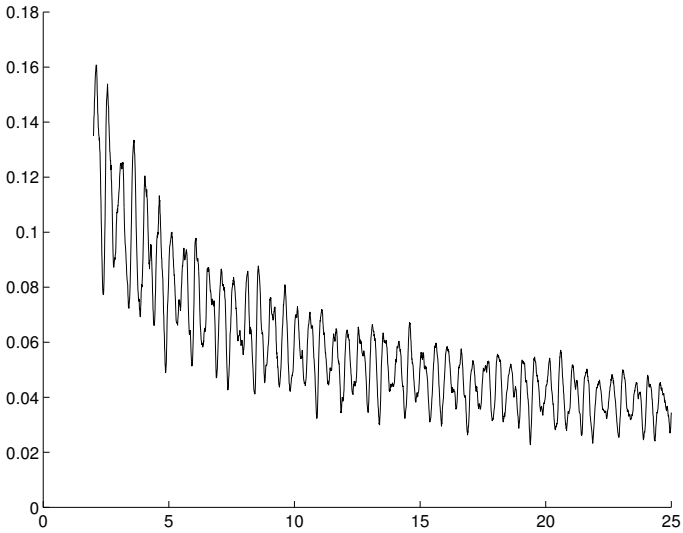


Figure 3.11: Quantification Errors on centroid estimation of a disk. Abscissa: Disk radius in pixels. Ordinate: Error on centroid estimate

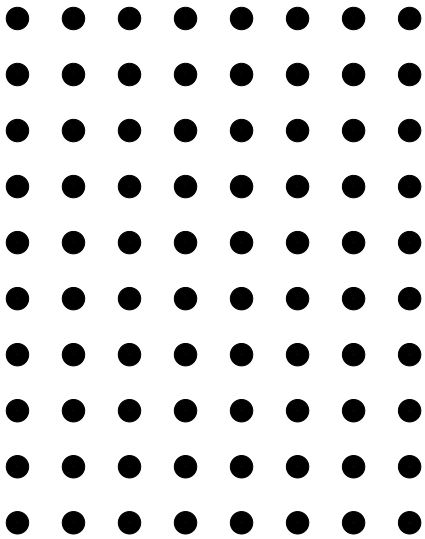


Figure 3.12: The calibration plane.

and changes in humidity. The doughnut shaped calibration objects almost resemble circles/disks (and will be referred to as such) as the center holes are very small and just big enough to enable the print of crosses to mark the object centers. Using a laser printer it is possible to place calibration objects with a pretty good precision, but still a greater precision is wanted for a better calibration. Therefore the center markings are placed to enable measurement of the center locations using other equipment, e.g. comparators. The calibration sheet is placed on a honed aluminum plate using a non-destructive glue.

A scaled version of the calibration plate is shown in figure 3.12. In the following both the single disk and the entire calibration plane is referred to as the calibration object. The meaning should be unambiguous from the context.

Two calibration plates were constructed. Each consisting of 80 circles in an array with a grid spacing of 25mm. One plate has got circles with a radius 10mm the other is constructed with circles of radius 5mm. As stated in section 3.5.4 there are contradicting arguments for the choice of disk size considering centroid estimation. The choice of a disk size of radius 5mm

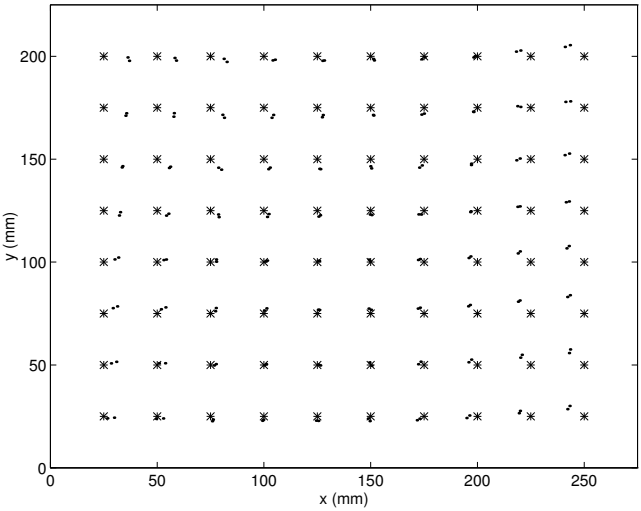


Figure 3.13: Comparative plot of nominal and true locations of calibration points. ∗: Nominal location. ·: Measured location relative to nominal location, distance enlarged 20 times. Measurements for 10mm and 20mm plate.

seems to be good choice in the actual case.

Though laser printers generally produce prints of great accuracy, the center locations are not considered to be found precisely at their nominal locations. Therefore the calibration plates were measured at the Institute of Surveying and Photogrammetry at the Technical University of Denmark using an optical stereocomparator.

The accuracy of the comparator is reported to be 0.005mm, which is significantly better than the true accuracy of the measurements which depend on the performance of the operator and the resolution of the laser printer. The overall accuracy of the center position measurements is estimated to be better than 0.05mm. In figure 3.13 the nominal locations are plotted. Small dots are plotted as well to illustrate the direction and magnitude of the errors of the actual measured points if they were to be located at the nominal locations. The error vectors are scaled with a factor 20 to make them visible. Data for both calibration planes are plotted. Since the coordinate system for the measurement of the actual locations is rotated and

	Max.error	Mean error	Std.deviation
Calibration plane 1 (10mm)	0.566	0.250	0.138
Calibration plane 2 (20mm)	0.603	0.226	0.145

Table 3.1: Errors(mm) on printer placements

translated with respect to the nominal coordinate system; the measurements have been translated back and attempted rotated into the nominal coordinated system.

Since the actual locations are known this investigation of the error vectors is not needed for estimating error on location measures, but solely for inspection of the performance of the laser printer. If the printer is printing systematically wrong or is doing unreliable placements, the shape properties of the printed objects can not be trusted (e.g. a circle becomes oval due to faulty paper transport or objects somehow become skewed). As it is seen, the errors are small and locally negligible when shape properties are considered. Though there seems to be a compression in the x-direction, meaning the x-translation of the paper is too slow. Also, the x-translation of the lower part of the paper (smaller y-values) seems to be quicker than the upper part. The compression in the x-direction is less than 1mm in 200mm.

Statistics on printer errors are found in table 3.1.

Chapter 4

Estimating the Axis of Rotation

When scanning objects all measurements are done in the same plane; the laser plane, and thus only produce sets of 2D measurements (actually 3D measurements with the same z-value: $z = 0$). 3D coordinate information is generated by rotating the object on the rotation table a known angle and then rotate the measured points the same angle but in the opposite direction out of the $z = 0$ plane. To do the 3D rotation around the axis of rotation, the axis is needed to be known or at least estimated. This chapter describes different approaches implemented for this purpose.

4.1 Methods to Estimate the Axis of Rotation

Four different approaches for estimation of the axis of rotation are described in the following sections:

1. Intersecting lines
2. Hough transform of lines
3. Intersection of virtual lines
4. Laser projection location

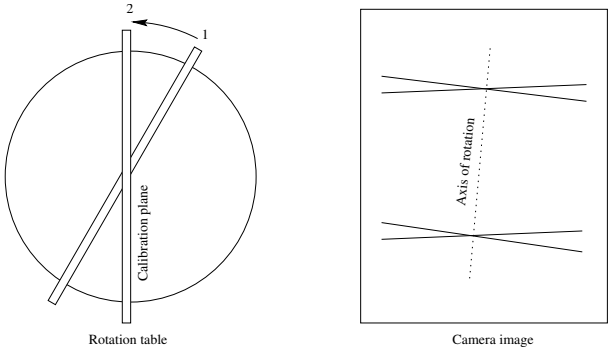


Figure 4.1: Principle of determination of rotation axis from the intersection of lines viewed at different rotation angles. Both views illustrated.

All estimation algorithms except the latter utilize the fact that the locations of points on the axis of rotation are invariant to the rotation.

4.1.1 Intersecting Lines

This procedure is based on the fact that lines are projected into lines by the perspective projection. Thus real world lines are found as lines in the image if the sensor system does not introduce any distortions. If the image of a line intersecting the axis of rotation then is regarded before and after a rotation, only one image point is found to be on the line in both images. This point is the intersection point of the line and the axis of rotation. Using two lines, two intersection points are estimated which is enough to define the axis of rotation. See figure 4.1.

Using more than two rotation steps and more than two lines, an overdetermined linear system is established for the estimation of the rotation axis. This method however will be inaccurate if the lens distortion is significant. The distortion will result in skewness of the line, making the estimate of a straight line inaccurate. One way to circumvent this is to use a pixel-based implementation of the method. With a calibration plate constructed with a number of lines printed with high contrast (see figure 4.2,right), images at different angles are simply thresholded at an appropriate level and then finally a logical AND is performed on the entire set of binary images. Only

Figure 4.2: Determination of rotation axis. Intersections found from thresholding the image frames followed by a logical AND of the frames.

foreground pixels on the rotation axis are found in all images(figure 4.2,left). Therefore the logical AND reveals in best case for each calibration line one pixel located on the axis of rotation. Nevertheless, the appropriate degree of rotation, the right threshold level, and the right line thickness are parameters hard to set. In some cases no pixel is revealed, in other cases a cluster of pixels is revealed. For clusters of pixels center of mass is used as an estimate of the intersection. In all cases location of the line intersection is never estimated with better accuracy than one pixel. Figure 4.2 shows an example of the pixel clusters and the resulting estimated axis of rotation found using this method.

4.1.2 The Hough Transform

The Hough transform is a well known and popular method for feature detection and extraction of geometric shapes in image analysis. Geometric shapes in image space are represented by parametric curves with a number of free parameters. The principal concept of the Hough transform is then to map the image space into a parameter space. For every feature point in

the image space, points in the parameter space with parameters defining a curve in image space, passing through the considered image point, are found. A set of accumulators or cells in the parameter space are defined and every time a parameter set corresponding to an image feature is found, the value of the accumulator is incremented. For binary images the increment is simply one, for grey tone images the grey tone value is used for increment. This is done for every feature point (or the entire image) and the corresponding parameters for every feature are found as local maxima in parameter space. If a certain number of features are searched for, the same number of local maxima are searched for in the parameter space.

Considering the case of line detection in images the lines may be parameterized as

$$y = ax + b$$

This representation has its limitations since vertical lines are not representable in this form. A more popular choice of parametric representation is the (ρ, θ) form:

$$\rho = xcos\theta + ysin\theta$$

However, in the case of calibrating the laser scanner, the orientation of the lines is known and the first more convenient form is chosen. From the line parameters found, the rotation axis is estimated in the same way as described in section 4.1.1.

An example of the calibration lines found deploying the Hough transform is shown in figure 4.3.

4.1.3 Intersection of Virtual Lines

The two methods described utilize another calibration pattern for the estimation of the axis of rotation than for the actual camera calibration. It would be convenient to use the same pattern for both processes. This is the motivation for the implementation of the virtual lines approach.

This methodology is based on the same properties of the perspective projection of a line as the two first methods. A line is completely described

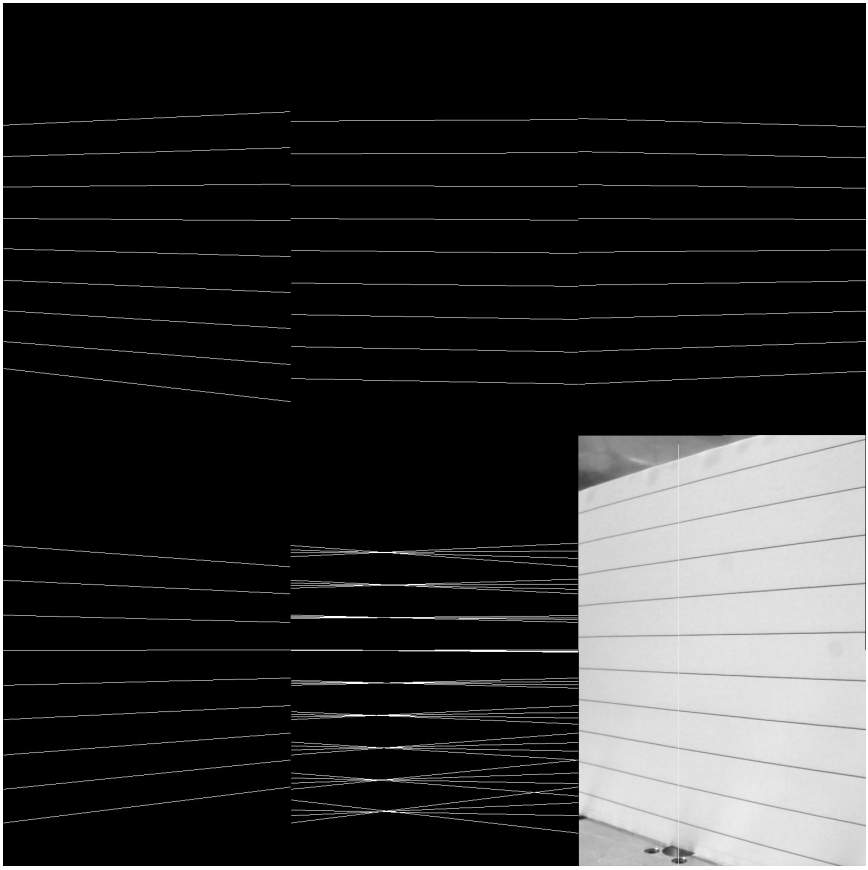


Figure 4.3: Determination of rotation axis. Hough transform approach. First 4 frames illustrate lines found at different rotation positions, frame 5, all estimated lines, and the the last frame shows the estimated axis of rotation.

from two points. Therefore it is possible to use two points known to be located on each side of the axis of rotation to describe exactly the same line as in method 1. This is easily accomplished using the camera calibration plane (see section (3.5.5) for description). When using a calibrated camera the located points are corrected for lens distortion and the virtual line connecting the points is calculated in undistorted image coordinates. Constructing virtual lines at different angles of rotation, the intersections of the lines are calculated in undistorted image coordinates. The resulting set of coordinates is then used to compute the real world coordinates for an estimate of the axis of rotation. An example of the implementation of this method is illustrated in figure 4.4.

It should be mentioned that using method 1, 2, and 3 to determine the axis of rotation, only lines which are not close to the optical axis should be used for the computations. If the calibration lines are close to a plane containing the optical axis, the perspective projection of the lines will be small in the direction perpendicular to the line. This is the feature utilized for generating intersections on the axis of rotation. The limited perspective distortion thus results in a less well defined intersection. This phenomenon is clearly seen in figure 4.2 and 4.4. For the actual setup only the three lines farthest away from the optical axis in both directions were used.

4.1.4 Laser Projection

The final method implemented to estimate the axis of rotation is quite straight forward. The camera calibration plane is rotated 90 degrees to a position perpendicular to the laser plane and the laser is turned on. The laser now illuminates the axis of rotation as the laser plane coincided with the calibration plane prior to the rotation. The location of the laser sheet projected upon the axis of rotation is then determined using the laser beam localization algorithm described in section 5.1.1.

4.1.5 Results on Estimating the Axis of Rotation

All four methods described above have been implemented for a comparative investigation of their performance on locating the axis of rotation. The first method was found usable but too inaccurate and unstable. The Hough transform did not produce the best result either. This method, however,

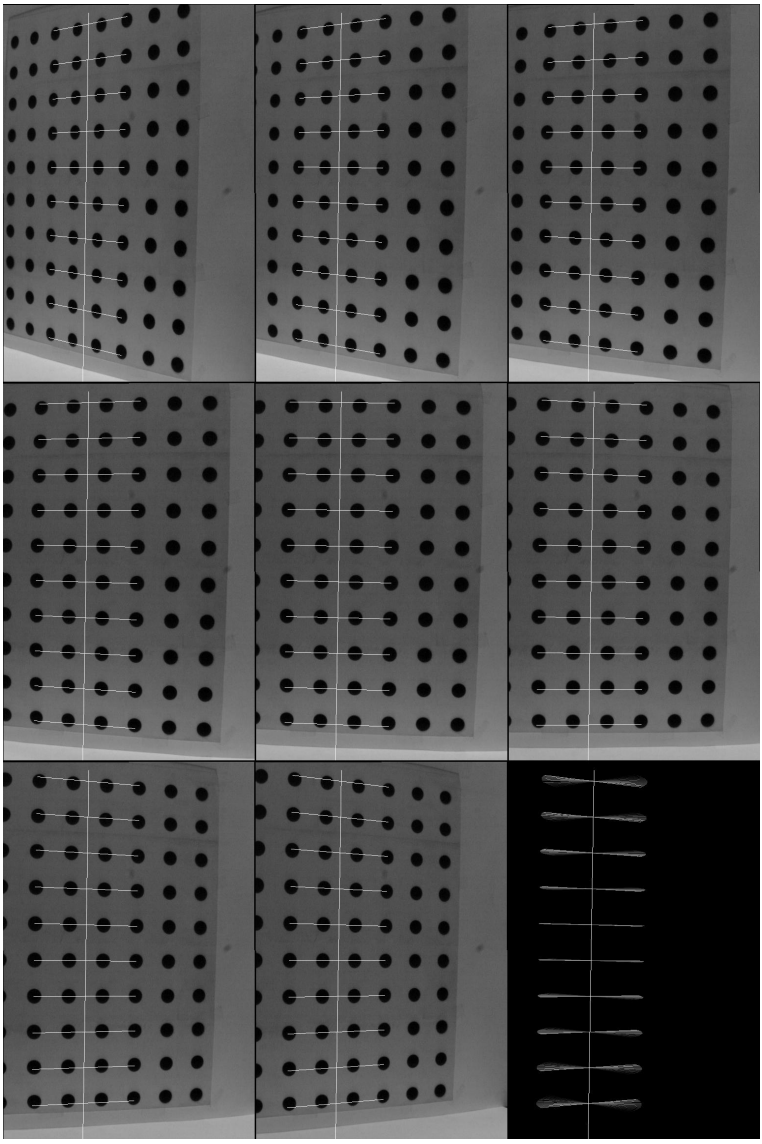


Figure 4.4: Determination of rotation axis. Virtual lines approach. First frames: Virtual lines at different rotation angles. Last frame: Estimation of axis of rotation.

may yield far better results if the image is resampled to undistorted image coordinates.

The last two methods produced good and robust estimates of the axis of rotation. The robustness of the estimation was checked as the reproducibility of the estimate. From doing repetitive estimates only very small deviations from one estimate to the others were found. It is harder to say anything about the accuracy of the estimate. The most reliable way to investigate the accuracy of the methods applied is to investigate the accuracy performance of the entire scanning system. Consult section 6.3 for an estimate of the overall performance. Another measure of the accuracy and robustness of the rotation axis estimate is to compare estimates from one camera with estimates from the other. Figure 4.5 is a comparative plot of estimates on the rotation axis using the virtual line approach and the detection of laser projection for both cameras. It is seen that all four estimates are within narrow limits and only deviate little from each other. The greatest distance among any of the estimates within the view of the scanner is found to be 0.5mm. If only the axis estimates made from laser projection are regarded, the maximal distance between the two estimates is 0.1mm.

The usage of the same calibration plane for determining the axis of rotation as for the camera calibration was found to be beneficial in implementing the final calibration algorithm. In the final implementation it was found, that locating the position of the axis of rotation from the projection of the laser sheet onto the calibration plane, revealed the best results.

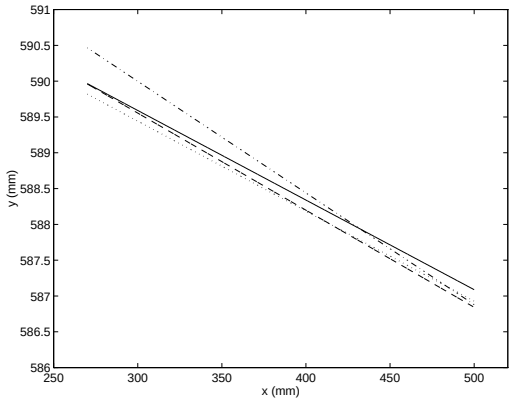


Figure 4.5: Comparative plot of rotation axis estimated for both cameras using different methods.
Solid line: Virtual lines approach, camera 1.
Dot-dashed line: Virtual lines approach, camera 0.
Dashed line: Laser projection, camera 0.
Dotted line: Laser projection, camera 1.

Chapter 5

Signal Processing and Data Handling

This chapter describes the methods deployed in the process of reforming the raw data sensed by the CCD-cameras to higher level data in the form of range images and 3D coordinate sets.

5.1 Signal Processing

The resolution of the CCD-chip is a limiting factor to the accuracy of coordinate measurements from the laser scanner. But the signal processing of the sensory information delivered by the CCD camera surely does influence the final result just as much, or even more. The correct signal handling may improve performance to become of subpixel accuracy, but assumptions on which the signal handling is based may also lead to erroneous results if real world conditions deviate too much from model assumptions. The following sections describe the signal processing and its basis as well as a number of errors likely to occur, and their influence on system performance.

5.1.1 Locating the Laser Beam

The signal of interest from the cameras is the reflectance of the laser beam on the object. As described, the laser light is beamed into a fan which is close to planarity. When projected onto an object, the beam appears as a line or stripe. This line is referenced to as the laser line.

As illustrated in figure 2.2 the intensity of the laser light along the length of the line is almost constant.

The intensity across the line however has a Gaussian distribution [1] [16], which constitute the basis of the algorithm for laser localization.

$$I(r) = I_0 e^{\frac{-2(r-r_0)^2}{{w_0}^2}} \tag{5.1}$$

I_0 is the maximum intensity, r_0 is the location (row index) of the max intensity. In laser technology literature the laser line width is measured at the $\frac{1}{e^2}$ points of the Gaussian distribution and thus is w = $2w_0$.

As discussed in section 3.5.4 the perspective projection does add some skewness to objects imaged by the cameras. However, within a local neighborhood (a few pixels) the difference between a perspective projection and an orthographic projection is limited. Assuming local orthographic projection, the laser signal sensed by the cameras will be Gaussian too if the beam is projected on a planar object with a constant reflectance coefficient.

Figure 5.1 illustrates a typical sampled version of the cross section, as seen by the CCD cameras.

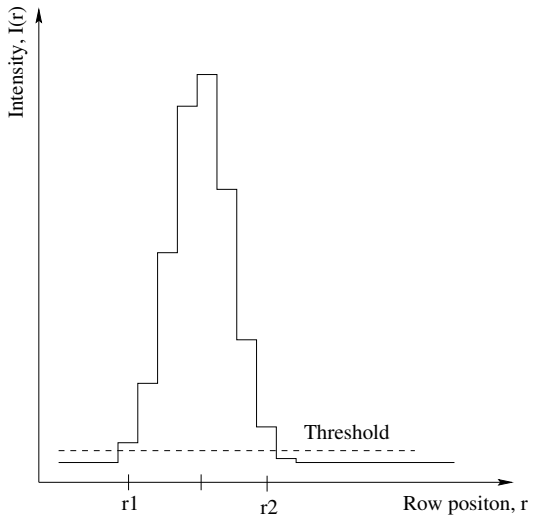


Figure 5.1: Laser line cross-section.

The aim of processing the signal is to estimate the center location of the laser line/beam. A number of methods are suggested.

Maximum intensity

In continuous space the center location simply may be deduced as being at the position of maximum laser intensity. In the discrete case this method is usable too, but the resolution will never be better than one pixel.

Interval midpoint

Instead of finding the maximum intensity, the image may be thresholded, and the center location of the beam is estimated, with reference to figure 5.1, as:

$$\hat{r_0} = \frac{r_2 + r_1}{2} \tag{5.2}$$

Moment based methods

Significant improvement to the resolution is obtained from deployment of moment based algorithms. Moment based methods are discussed in detail in section 3.5.2. Deploying the center of gravity calculation for the estimation of the beam center position is of course only legitimate due to the symmetric properties of the Gaussian distribution. The one dimensional case of a center of gravity (mean) estimate reveals:

$$\hat{r_0} = \frac{\sum_{r=0}^{N_r-1} rI(r)}{\sum_{r=0}^{N_r-1} I(r)} \tag{5.3}$$

N_r being the total number of rows, r the row number, and I(r) the intensity at position r.

Including a previous thresholding of the image (5.3) becomes:

$$\hat{r_0} = \frac{\sum_{r=r_1}^{r_2} rI(r)}{\sum_{r=r_1}^{r_2} I(r)} \tag{5.4}$$

Curve fitting

Based on the assumption of a Gaussian intensity profile, it is possible to do an ordinary curve fitting on the sampled data with a following derivation of the peak position. This method, however, will require well-behaved surfaces, i.e. geometrically smooth variating surfaces without sudden changes of the reflectance coefficient.

Implementation

Constructing the laser scanner, the moment based approach with a preceding thresholding, was chosen. The choice of this algorithm was due to its robustness and the property of subpixel accuracy. The width of the laser beam is typically 5-7 pixels at the current setup which enables exploitation of the subpixel performance of the algorithm.

Even when using the scanner in complete darkness, it is necessary to do a preceding thresholding of the image to avoid the influence from dark current

in the CCD-chip. The threshold level should be kept as low as possible to enhance the performance of the center estimate algorithm. With increasing ambient lighting the threshold level of course needs to be raised, and at the point of higher reflectance of light originating from other sources than the laser, the algorithm fails.

The algorithm is implemented as a search of each column registrating all intervals with intensity values above the threshold. The choice of the right interval may be done from different criteria: The interval with the highest peak, the interval holding a higher 'mass' (the sum of the intensity values), or the interval attributed by the highest 'density' (the sum of intensity values divided by the number of pixels).

Assuming no ambient lighting stronger than the laser, the natural choice was to use the interval with the highest peak value. The intensity values of the interval is then used to estimate the beam center location from (5.4). The peak value is stored for possible texture mapping.

5.1.2 Error Sources

As described in the previous section, the performance of the signal processing will suffer at conditions deviating from perfect conditions.

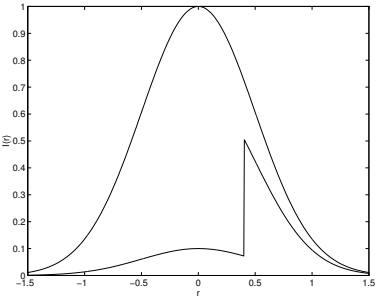
A rough surface may introduce laser speckle resulting in multiple peak detection. The same is likely to happen for metallic surfaces or surfaces with specular reflection. Those kind of errors are limited by the investigation of multiple intervals as described above.

In section 2.5.1 the errors due to occlusion effects are described. In some cases the occlusion will only be partial, resulting in not a missing signal but a partial signal which will lead to an erroneous peak location estimate.

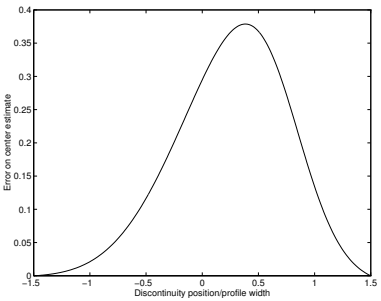
Also if the the object deviates from planarity, the crosssection of the laser profile will be smeared or skewed.

Even if the object is perfectly plane a change in the reflectivity of the object will deform the profile. Figure 5.2(a) illustrates the Gaussian profile of the perfectly reflected beam (algorithm assumption) and the reflection from an object with a reflection coefficient stepping from 0.1 to 0.7. The beam is rescaled to the width w $= 2w_0 = 2$ and the discontinuity is located at $0.4w_0$. As seen in figure 5.2(b) $0.4w_0$ is the discontinuity location resulting in the maximal error of this kind. Both figures are at a view perpendicular

to the surface, and the error of $0.38w_0$ is increased to approximately $0.5w_0$ at the gazing angle in the neighborhood of the laser scanners viewing angle.



(a) Perfect laser profile and profile distorted from varying reflection coefficient.



(b) Error on center estimate vs. reflectance discontinuity position. All measures scaled with w

Figure 5.2: Errors due to variating object reflectivity

Regarding even higher differences of the reflection coefficients, the errors of course will increase. The problem may be reduced by painting the object, but this is only a possible solution in special cases and will change the measures of the object. In most cases median filtering the range image will improve results. In the case of measuring sugar beets, the variance of the reflectivity of the objects is limited and the problem is not addressed. Curless and Levoy [5] have some interesting approaches to reduce the problem by analyzing the time evolution of the laser signal.

Figure 5.3 is a good example of range errors due to changes of the reflectivity properties of the object surface. The images are blow-ups of data from the scan of a plastic oil can. The left frame shows the range data and the right frame shows the laser band reflectivity in the same area. The grey tone values of the images are scaled to give a better visual impression of the phenomenon. However, the influence of the changing reflectance is evident: The range should variate smoothly, but the text (lower reflectivity) is clearly marked in the range data. In figures, the error introduced from changing reflectivity has a magnitude of approximately 0.5mm.

Figure 5.3: Error due to reflectance discontinuity. Left: Range image. Right: Laser reflectance.

5.2 Data Handling

Scanning an object may result in a set of 500000 or more points with corresponding intensity information. Those coordinate sets are irregularly distributed. To be able to do rendering and calculate object properties from this huge collection of data; data needs somehow to be arranged. One solution is to arrange the point collection by applying a Delaunay triangulation. Another approach is to take advantage of the regularity already found in the point collection and then produce a regular grid of data from interpolation. The latter approach was adapted in this project for a number of reasons:

- Regular point sets are easier to handle
- Merging data from two cameras into one set of data points may become a difficult and cumbersome task using irregularly distributed points.
- Comparisons on the performance of one camera versus another is easily done from regularly distributed points.
- General 2D image analysis methodologies are often applicable to regularly distributed point sets.
- A regular grid of data is conveniently visualized in 2D as well as 3D

5.2.1 Range Imaging

Image formation

For every step of the in total 360 degrees rotation, 768 coordinate sets are found (some may be marked *void*). Coordinates are either delivered as cylindrical coordinates (r, θ, h); the range r measured as the perpendicular distance to the axis of rotation, the angle θ the negated angle of rotation of the rotary table, and h the projection of the point onto the axis of rotation. Or the coordinates are directly transformed to cartesian coordinates (x, y, z).

As every rotation step has got the same angular increment some regularity in the distribution of data points already exists. This regularity is used to generate range images. The range image is structured in a $N_p \times N_s$ grid, where N_p is the maximum number of points acquired at each step angle (usually 768), and N_s is the number of scans (steps) for a complete

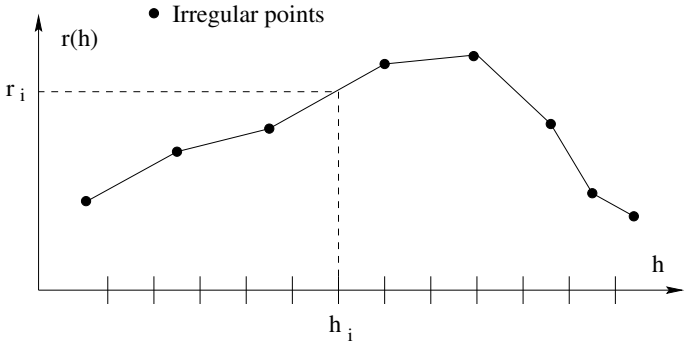


Figure 5.4: Interpolating range values at regularly distributed h-values.

scan (typically 362 or 724). The grid spacing from one row to the next is determined by the overall maximum and minimum values of h:

$$\delta h = \frac{max(h) - min(h)}{N_p}$$

The spacing in between columns is of course given by the angle of rotation.

All 768 points collected at a particular angle are sorted according to their h-value. Actually, in most cases the points are already ordered according to their h-values if the object scanned is not hull. The points now constitute the basis for a function $r(h)$ which is used to interpolate r-values for all cells in the grid for this particular column. The linear algorithm is illustrated in figure 5.4.

For some objects, however, there is not a one-to-one correspondence from h to r. By for example scanning a tea cup, the scanner might 'look' down into the cup and register the inside of the opposite side. This will result into two r-values for the same h-value, and the description of the range as a function of the height is thus impossible.

Scanning sugar beets this is not a problem.

Repeating the process for all angular positions, a complete range image is produced (see e.g. figure 5.5).

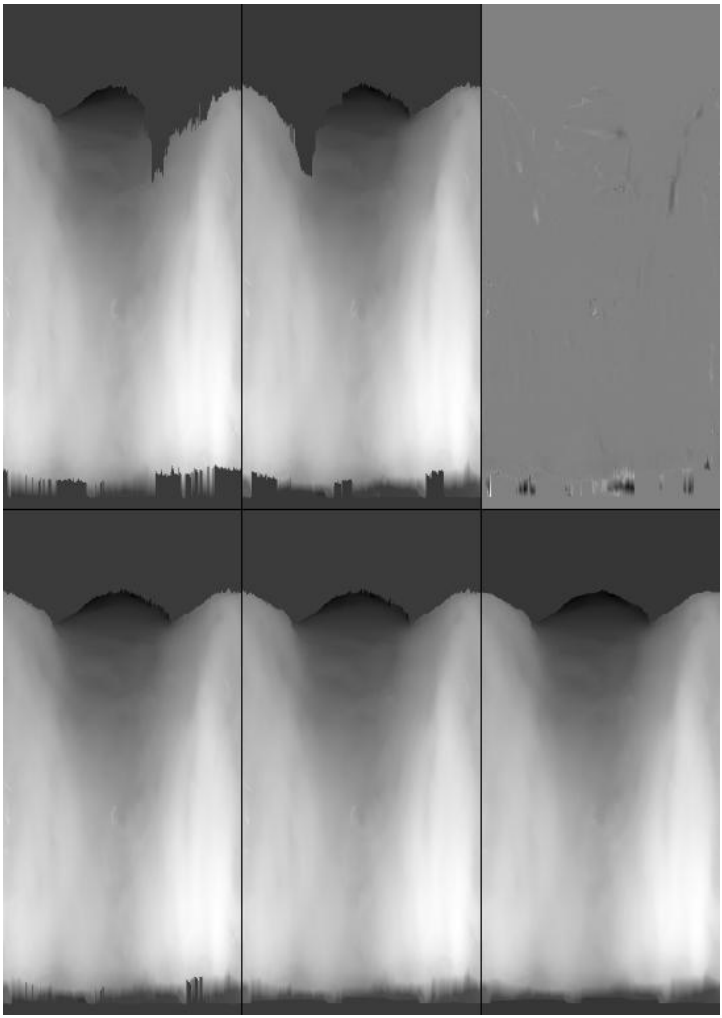


Figure 5.5: Merging data from two cameras (beet 26). Frame 1: Range data, camera 0. Frame 2: Range data, camera 1. Frame 3: Difference (frame 1 - frame 2). Frame 4: Merged data. Frame 5: Filtered. Frame 6: Median filtered.

Data merging

The merge of the data of the two range images is far easier handled using the regularly distributed points than using the irregularly distributed point set. One approach is to simply fill out the areas with data missing from occlusions in one picture with data from the other. However, there will always, due to calibration inaccuracy, exist a general difference from the data acquired by one camera and the data acquired with the other. Therefore enhancement should be done, not only in areas with missing data, but in the entire image. This problem is addressed by calculating the range difference image as:

$$D_r(h,\theta) = \begin{cases} r_1(h,\theta) - r_0(h,\theta) & r_0(h,\theta) <> void, r_1(h,\theta) <> void \\ 0 & Otherwise \end{cases} \tag{5.5}$$

r_1 being range data from camera 1 and r_0 the range data from camera 0. The third frame in figure 5.5 shows an example of the range difference image calculated in this manner. Now, one way of merging the data is to use the mean value if data are found in both images, and use adjusted values from one image in areas where only data from one image is available:

$$r_m(h,\theta) = \begin{cases} r_0(h,\theta) + \frac{1}{2}D_r(h,\theta) & r_0(h,\theta) <> void, r_1(h,\theta) <> void \\ r_0(h,\theta) + \frac{1}{2}C_d & r_0(h,\theta) <> void, r_1(h,\theta) = void \\ r_1(h,\theta) - \frac{1}{2}C_d & r_1(h,\theta) <> void, r_0(h,\theta) = void \end{cases}$$

where C_d has been assigned an appropriate value, e.g.

$$C_d = Mean(D_r)$$

The result of the data merge is visualized in frame 4 of figure 5.5.

Intensity information is handled analogously and the process is illustrated in figure 5.6.

Scanning Sugar Beets

In most cases some post processing of the data needs to be done. One thing that needs to be done in almost any case, is to separate the object

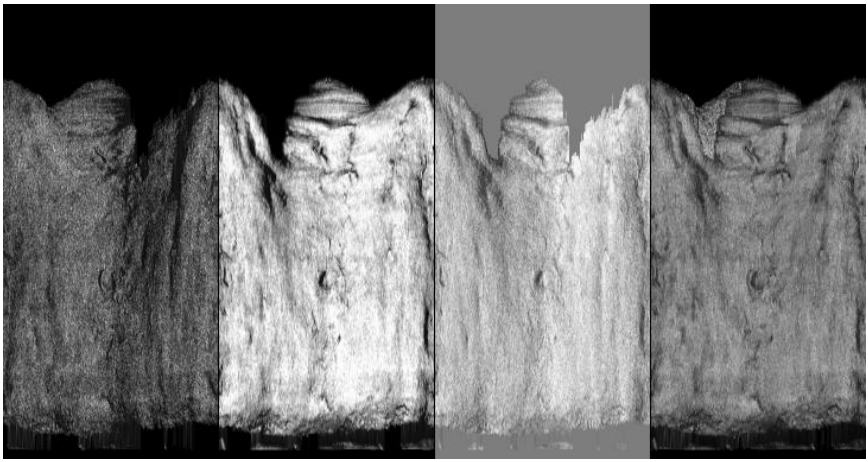


Figure 5.6: Merging intensity data from two cameras. Frame 1: Laser reflectance, camera 0. Frame 2: Laser reflectance, camera 1. Frame 3: Difference image. Frame 4: Merged intensity data.

from the platform on which it was placed for scanning. In many cases this is a matter of dropping all data found below a certain h-value, but in the process of scanning sugar beets the problem is a bit trickier. The sugar beets are for fixation purposes placed on a 3-winged spike , which is partly outside the beet and therefore visual to the scanner. It is not an easy task to separate the two objects, but some ad hoc based methods have been investigated.

One idea is to disallow steady values of $\delta r/\delta h$ for r-values below a certain value (the width of the spike). I.e. data points with r-values which are smaller than or close to the width of the spike, are candidates to be points on the fixation device. If neighboring points in the direction of increasing or decreasing h hold r-values with only small deviation from each other, the whole set of points is assumed to represent the fixation device and is therefore discarded. Holes generated from discarding data in this manner are, if there exist valid data at the particular h-level in the neighborhood, filled by interpolation. Frame 5 in figure 5.5 illustrates the result of the approach. Most of the fixation device had though already been removed in this case by simply restricting the validation interval for h.

Another problem scanning sugar beets are the small hairy roots sticking out from the body of the beet (see e.g. figure A.30). Those are occluding the actual body of the beet and are untreated assumed to be part of the body. Their appearance will add errors to calculations of volumetric properties, and they are therefore unwanted.

It was found, that the most common orientation of those tiny roots is the same as the direction of increasing/decreasing h-value. This fact motivated the employment of a horizontal median filtering of the range image to remove the roots. Filtering was performed as a wrap-around filtering at the edges which is natural due to the nature of the image. The size of the median filter should be set according to the angular step size of the data. At 362 steps/rotation a filter size of 9-11 was found appropriate. Median filtering produced fine results. The median filtering will also be helpful if noise generated spikes are found in the data. This may e.g. occur when using an extremely low threshold for the acceptance of data in the laser centroid estimation. In such cases dark current in the CCD-cameras may lead to faulty spike-like errors.

Frame 6 in figure 5.5 shows the final result after median filtering and a 3D-rendering of the same data is found in figure 5.7

Figure 5.7: 3D-rendering of the range data from figure 5.5

Chapter 6

Performance

6.1 Introduction

This chapter investigates the performance of the scanner system. A separate performance investigation is applied to the calibration routines.

Points	Method	Mean error	Std.dev.	Max error
78	DLT	0.553	0.286	1.72
39	DLT	0.661	0.533	2.58
20	DLT	0.686	0.493	2.43
10	DLT	0.686	0.557	3.09
78	Tsai	0.052	0.030	0.16
39	Tsai	0.072	0.039	0.22
20	Tsai	0.065	0.032	0.15
10	Tsai	0.077	0.037	0.18
78	Tsai-3D	0.046	0.029	0.16
39	Tsai-3D	0.040	0.020	0.09
20	Tsai-3D	0.041	0.021	0.09
10	Tsai-3D	0.045	0.027	0.11

Table 6.1: Errors on backprojection of calibration points on 10mm calibration plane. Calibration methods: Direct Linear Transformation and Tsai's two stage. When 78 points are used for calibration, calibration points and points used for error measurements are the same. The same applies to Tsai-3D measurement (see text for explanation). All other error measurements are calculated on a separate point set. All measurements in mm

6.2 Calibration Performance

To investigate the performance of the calibration routine the calibration data was used to backproject the image points corresponding to the calibration objects. An error on the estimate was then calculated in comparing the estimate with the true value. Using the calibration points as test points for the performance measure of the calibration may reveal wrong impressions of a well performing system since the object-camera transform may do a perfect fit on the calibration points but a sloppy fit elsewhere. Though this phenomenon is much more pronounced in for instance a high order polynomial warp model, which does not model the nature of the setup (distortion, projection, viewing angle, etc.), than for a model based on real world phenomena, it still may influence the performance measure. It is therefore better to utilize a separate set of test points for measurement of the performance. The results are tabulated in table (6.1) and (6.2)

The error measurements of type DLT and Tsai are measured as the error on the true location of the calibration point and the location of the intersection

Points	Method	Mean error	Std.dev.	Max error
78	DLT	0.555	0.291	1.83
39	DLT	0.605	0.400	2.31
20	DLT	0.615	0.349	2.10
10	DLT	0.663	0.494	2.85
78	Tsai	0.058	0.034	0.16
39	Tsai	0.056	0.031	0.11
20	Tsai	0.063	0.039	0.16
10	Tsai	0.073	0.045	0.19
78	Tsai-3D	0.048	0.028	0.15
39	Tsai-3D	0.046	0.026	0.11
20	Tsai-3D	0.048	0.032	0.13
10	Tsai-3D	0.047	0.035	0.14

Table 6.2: Errors on back projection of calibration points on 20mm calibration plane. See table 6.1.

of the calibration plane with the back projected ray. The back projected ray is constructed as the ray through the optical center and the undistorted image coordinates. The Tsai-3D measure indicates the perpendicular distance from the true location of a calibration point to its corresponding back projected ray. It is therefore always less than or equal to the in-plane distance measure.

Due to the nature of the laser scanner where all measurements are done in the laser plane coinciding with the calibration plane, the Tsai-3D measure has limited interest.

From table 6.1 and 6.2 it is seen that the number of calibration points only has a limited effect on the performance of the camera calibration. The size of the calibration objects (the diameter of the disks) does not seem to have any significant influence on the performance of the algorithms. As stated in section 3.5.4, larger size objects are expected to add more error to the moment based centroid estimate due to their more pronounced perspective projection. The fact that this phenomenon does not come clear in the error measurements is probably explained by the nature of the method for doing the error measurement: Both the estimation of the location of the calibration points and the test points is carried out using the moment based centroid estimate.

It is however clear, that the usage of a model including underlying error sources, significantly improves the performance. In general the Tsai model performs about a factor 10 better than the direct linear transformation and its implementation is therefore justified.

6.3 Scanner Performance

The performance of the laser scanner is described from a number of properties: The primary measure of performance is the geometric accuracy of the object coordinates obtained. Secondly, speed and robustness of the system are properties of great appreciation.

Speed has not been a major issue in this project. A complete scan using two cameras including intensity information (in the laser frequency band) is done in approximately 5 minutes at the current setup. The time usage for a scan is easily reduced using some of the features described in section 2.3.

6.3.1 Geometric Accuracy

To investigate the accuracy of the laser scanner, two test objects of known dimensions were measured for comparison of measured coordinates with true dimensional properties of the objects.

Plane Measurements

One planar test object was used for measuring the accuracy. The test plane was 30mm by 100mm in size and every measurement included approximately 18000 points. For every test a plane was estimated using the measured coordinate triplets, and the distances from the measured points to the estimated plane were calculated to estimate the accuracy of the measurement. The results of the test are tabulated in table 6.3.

The standard deviation on the distance from the measured points to the fitted plane is seen to be less than 0.05mm, which is quite excellent for a system of low cost. This method of accuracy estimation does however include less appreciated attributes. For one, the area of the planar test object is limited. Also, the test object was due to construction limitations limited to a narrow band of positioning (close to the axis of rotation).

An interesting observation is that the usage of the Tsai camera calibration does not result in much better results than the implementation of the direct linear transformation. It is difficult to find any other explanation to this than the fact, that coordinate measurement are performed within a very

Camera	Mean error	Std.dev.	Max error	Method
0	-0.000082	0.0498	0.243	Tsai
1	-0.000050	0.0392	0.246	Tsai
I	-0.000032	0.0310	0.124	Tsai
0	-0.000111	0.0580	0.268	DLT
1	-0.000078	0.0485	0.291	DLT
I	-0.000060	0.0426	0.212	DLT

Table 6.3: Errors on plane measurement. Camera I is the interpolated data derived from both cameras. All figures in millimeters.

limited area, and the direct linear transformation is apparently doing a good fit in this area, whereas the overall fit to the entire laser plane is worse (see section 6.2).

It is also observed that camera 1 in general is performing better than camera 0. Camera 1 is newer than camera 0 and this may be a factor influencing on the difference in performance of the two cameras. Nevertheless, a possibly even more significant factor is the fact that the camera lenses are set at different apertures. Prior to calibration the apertures were set to obtain good signal dynamics. Beaming the laser on the object to be measured, the apertures were set such that the maximum intensity sensed was just below dynamic limits of the CCD-camera. For some reason camera 0 always output higher maximum values than camera 1, and hence a smaller aperture was used for camera 0. This may have been due to single malfunctioning sensor elements of the CCD array, and it might have been better to ignore a few overflowing outputs to get better dynamics in the image.

Using the planar test object does, due to the limited degree of rotation, only include a limited ability to detect an erroneous estimation of the axis of rotation. Therefore the performance has been tested deploying an object which enabled full rotational scans.

Complete rotation tests

For test purposes a steal cylinder with a known diameter (84.95mm) was scanned. Slicing the cylinder perpendicular to the axis of symmetry reveals a circular profile of the same diameter. The performance is then measured from the errors on the measured positions as if they were to be located on

the circle. Figure 6.2 shows a yz-plot of all points measured within the x-interval [463mm;464mm]. For this particular slice the standard deviation of the coordinate measurements was 0.0495mm and the maximum error was 0.24mm. The error measurements were repeated along the axis of symmetry and the overall average was calculated. Results are found in table 6.4.

Again there is no significant difference observed from using one method instead of another. The same explanation as for the test using the plane may explain this: Apparently there is a very good fit to the direct linear transformation within this particular area of measurement. The accuracy of the measured points is far better than for the back projection of calibration points (section 6.2)!

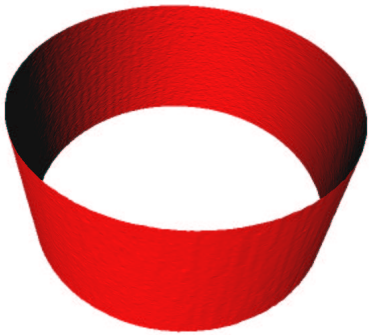


Figure 6.1: 3D rendering of scanned cylinder

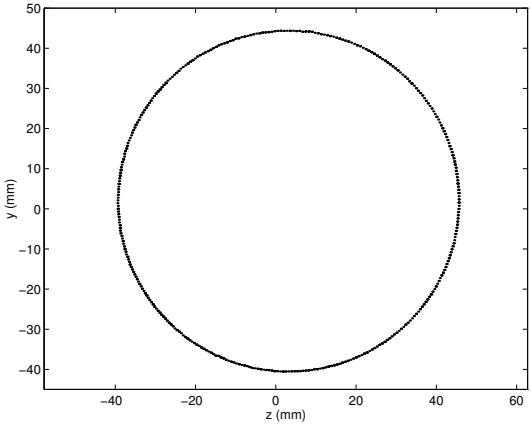


Figure 6.2: A slice of the test cylinder. Standard deviation on coordinate errors: 0.05mm. Max.error: 0.24mm

Camera	Mean error	Std.dev.	Max error	Method
0	0.06	0.12	0.53	Tsai
1	0.05	0.05	0.26	Tsai
I	0.04	0.04	0.21	Tsai
0	0.03	0.12	0.58	DLT
1	0.06	0.05	0.25	DLT
I	0.03	0.04	0.22	DLT

Table 6.4: Errors on measurement of the test cylinder. Camera I is the interpolated data derived from both cameras. All figures in millimeters.

Chapter 7

Object Shape Properties

This chapter describe the calculation of some basic geometric properties which are utilized for a preliminary investigation of the shape of sugar beets.

7.1 Basic Dimensional Properties

Volume

Using the dataset produced by the interpolation process described in section 5.2, the volume of a scanned object is easily calculated. All coordinates are ordered in slices with the same x-value, and the volume is thus calculated by multiplying the area of every slice with spacing in between the slices (the x-increment):

$$\begin{aligned} V &= \sum_{i=0}^{N_s-1} \delta x A_i \\ &= \delta x \sum_{i=0}^{N_s-1} A_i \end{aligned}$$

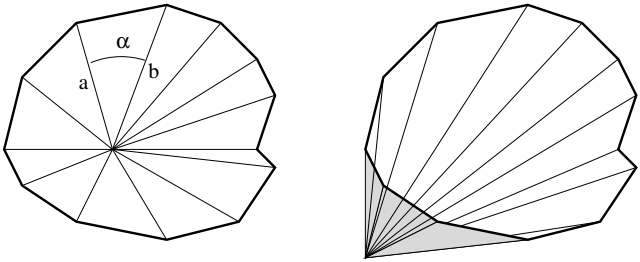


Figure 7.1: Area of a polygon

where N_s is the number of points describing the polygon which is the same as the number of rotational steps during the complete scan. The area of every slice is calculated from the general area calculation of polygon areas. The polygon is subdivided into triangles. With reference to figure 7.1 triangle j is seen to have the area:

$$A_j = \frac{1}{2} a_j b_j \sin \alpha_j$$

The intersection of the axis of rotation and the polygon slice is the natural choice of the base point common to all triangles. However, some object slices do not include the rotation axis. As illustrated in the right part of figure 7.1 this is no problem: Since the points describing the polygon are traversed in order, α is negative (positive angle in anti-clockwise direction) for the shaded triangles, and the area of those triangles is subtracted when doing the summation of the total area.

In total, the volume calculation becomes:

$$V = \delta x \sum_{i=0}^{N_p-1} \sum_{j=0}^{N_s-1} A_{ij}$$

N_p being the number of slices.

Center of Gravity

Assuming a homogeneous density of the object, the center of gravity is calculated for each of the triangles described above. Every triangle is assigned the 'volume' ($A_j \delta x$) and the center of gravity for the entire object is then calculated by summing all triangle center vectors weighed by their individual 'volume'-fraction of the total volume.

7.2 Principal Components

For an investigation of the volume distribution of an object, in particular a sugar beet, a principal component analysis was adopted.

Principal components

Considering the three-dimensional coordinate triplet for the position of a volume unit as a stochastic variable

$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

a complete object is described as all observations of $\mathbf{X}$ which are part of the object.

Estimating the objects dispersion matrix $D(X) = \mathbf{\Sigma}$ it is possible to apply a principal components analysis to get an idea of spatial distribution of the object. The directions of the principal axes of the object are found from the eigenvectors of $\hat{\mathbf{\Sigma}}$ and the eigenvalues λ_1, λ_2, and λ_3 are measures of the eccentricity of the object.

Thus the following shape descriptors are formed:

$$D_1 = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}$$

$$D_2 = \frac{\lambda_2}{\lambda_1 + \lambda_2 + \lambda_3}$$

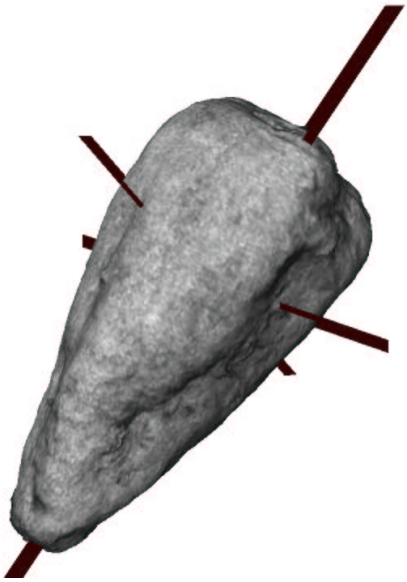


Figure 7.2: Principal axes, Beet 26.

$$D_3 = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}$$

The eigenvalues and their corresponding eigenvectors are ordered as $\lambda_1 > \lambda_2 > \lambda_3$, and the eigenvectors $\mathbf{p_1}, \mathbf{p_2}$, and $\mathbf{p_3}$ are referenced as the primary, secondary, and tertiary principal axis. The i'th principal component of $\mathbf{X}$ is the projection $Y_i = \mathbf{p}_i^T \mathbf{X}$. And the vector of principal components is [4]:

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \mathbf{P}^T \mathbf{X}$$

$$\mathbf{P} = [\mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3]$$

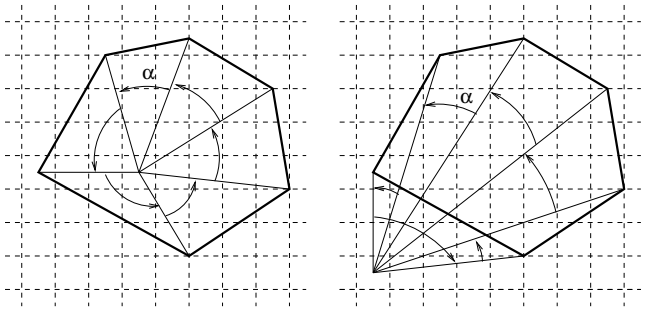


Figure 7.3: Voxel generation from object slice tessellation. Left: Cell inside polygon, $\sum \alpha_i = 2\pi$. Right: Cell outside polygon, $\sum \alpha_i = 0$.

Volume tessellation

To enable the possibility of calculating principal components, the scanned objects need another representation. For this purpose a voxel based model is created from the set of irregular points. A voxel based model is constructed by placing a grid over every slice (described in section 7.1). Every cell formed by this tessellation is then determined to be either inside or outside the polygon describing the boundary of the slice. If the cell is inside the polygon, a voxel is constructed by adding a third dimension with the height δx to the cell and the voxel is marked as being part of the object.

To determine if a cell is inside the polygon, two method are applicable. One method is to draw a line from the cell center to a far away point and calculate the number of intersections of this line and the polygon border. If there is an odd number of intersections, the cell center is inside the polygon - otherwise outside. This method is subject to numerically instability if the line is crossing one of the polygon segments close to parallel; therefore the second method was chosen for implementation. The second method is illustrated in figure 7.3.

Considering a particular cell, α is calculated for all the triangles delimited by two neighboring vertices on the polygon and the cell center. The sum of all α's will tell if the cell center is inside or outside the polygon. An angle sum of 2π tells that the cell center is inside the polygon. If the center is outside, the angles will sum to zero.

From a voxel based model the estimation of the dispersion matrix $\mathbf{\Sigma}$ is

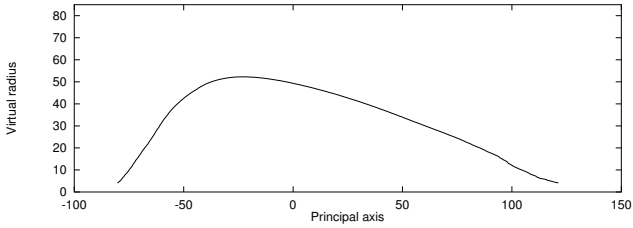


Figure 7.4: Virtual radius of the 'standard' beet.

trivial, and from that the principal axis and components are calculated.

7.2.1 Shape Properties of Sugar Beets

70 sugar beets were scanned for an investigation of their shape properties and spatial distribution. The measures D_1, D_2, and D_3 described in section 7.2, were all calculated and the length of the primary principal axis was estimated.

Also, the volume was projected on the primary principal axis and then transformed to the equivalent radius, i.e. a virtual radius which would be the perpendicular distance to the object surface if the mass was evenly distributed around the primary principal axis.

All coordinate sets of the 70 sugar beets were translated into Open Inventor format including texture mapping for the purpose of rendering.

The calculated properties are summarized in table 7.1 and table 7.2. Statistics are found in table 7.3.

All data is found in appendix A, and it is interesting to see how there seems to be different types of shape of beets.

A 'standard-type' of sugar beets was attempted estimated by calculating the mean of the virtual radius of all beets. The result is seen in figure 7.4 and the standard deviation of the estimated mean is visualized in figure 7.5.

The estimate agrees with the mean length of the primary principal axis, which in table 7.3 is found to be 188mm. This model, however is modelling the beet as being symmetrical with respect to the primary principal axis. This assumption seems not to hold if the parameters D_2 and D_3 are inspected. If this symmetry was true, D_2 should be equal to D_3. From the

Beet	D_1	D_2	D_3	l_1	Volume
1	0.623	0.208	0.169	222	1015
2	0.515	0.260	0.226	174	1068
3	0.510	0.259	0.230	164	661
4	0.589	0.232	0.179	169	613
5	0.621	0.208	0.171	155	643
6	0.538	0.264	0.198	176	1051
7	0.660	0.192	0.149	223	959
8	0.558	0.261	0.181	176	1241
9	0.695	0.173	0.132	227	811
10	0.554	0.240	0.205	169	1229
11	0.615	0.210	0.175	235	1250
12	0.649	0.195	0.156	213	858
13	0.654	0.192	0.153	220	941
14	0.593	0.240	0.168	180	614
15	0.617	0.210	0.173	197	766
17	0.581	0.228	0.191	196	982
18	0.585	0.220	0.195	174	594
19	0.561	0.250	0.188	185	951
21	0.598	0.217	0.185	238	1554
22	0.600	0.226	0.174	194	839
23	0.624	0.200	0.176	204	705
24	0.644	0.188	0.168	184	591
25	0.633	0.201	0.166	152	301
26	0.651	0.200	0.149	183	693
27	0.639	0.188	0.174	164	569
28	0.531	0.256	0.213	149	551
29	0.643	0.193	0.164	234	1042
30	0.479	0.269	0.252	145	851
31	0.451	0.291	0.257	183	1497
32	0.646	0.182	0.171	190	689
33	0.686	0.170	0.144	181	523
35	0.722	0.149	0.129	199	654
36	0.602	0.219	0.179	222	1358
37	0.694	0.171	0.134	194	493
38	0.635	0.206	0.159	177	688
39	0.667	0.179	0.154	186	747
40	0.585	0.220	0.194	214	948

Table 7.1: Shape properties of scanned beets.

Beet	D_1	D_2	D_3	l_1	Volume
41	0.704	0.155	0.142	187	684
42	0.515	0.262	0.222	173	1125
43	0.636	0.200	0.164	195	974
44	0.510	0.268	0.221	174	1256
45	0.560	0.261	0.179	172	909
46	0.640	0.202	0.159	203	960
47	0.583	0.223	0.194	175	1028
48	0.547	0.257	0.196	174	964
49	0.467	0.310	0.222	162	1053
50	0.714	0.161	0.125	174	475
51	0.654	0.194	0.152	214	1379
52	0.576	0.220	0.204	167	711
53	0.581	0.242	0.176	174	993
54	0.604	0.211	0.184	192	828
55	0.538	0.250	0.212	176	1020
56	0.650	0.185	0.165	232	1436
58	0.646	0.187	0.167	198	921
59	0.504	0.262	0.234	158	909
60	0.641	0.206	0.153	186	774
61	0.607	0.225	0.168	174	830
62	0.667	0.200	0.134	181	709
63	0.381	0.340	0.279	148	482
64	0.630	0.219	0.151	179	783
65	0.514	0.264	0.222	159	678
66	0.667	0.176	0.157	199	1067
67	0.706	0.158	0.135	226	1046
68	0.690	0.165	0.144	197	821
69	0.617	0.224	0.160	232	863
70	0.585	0.228	0.187	218	1644
71	0.612	0.210	0.178	190	1121
72	0.601	0.211	0.187	192	751
73	0.603	0.209	0.188	168	692
74	0.641	0.180	0.178	197	855
75	0.504	0.282	0.214	174	930

Table 7.2: Shape properties of scanned beets.

	D_1	D_2	D_3	l_1	Volume
Mean	0.602	0.219	0.179	188.3	890.3
Standard deviation	0.068	0.038	0.031	23.3	271.5
Max	0.722	0.340	0.279	238	1644
Min	0.381	0.149	0.125	145	301
Median	0.612	0.211	0.175	184	858

Table 7.3: Statistics on shape properties of scanned beets.

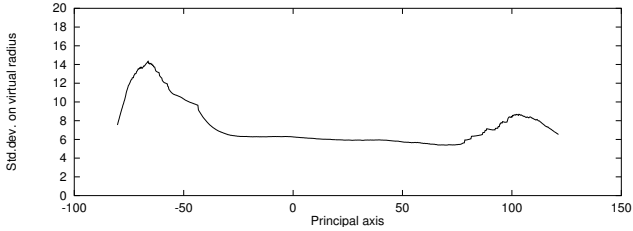


Figure 7.5: Standard deviation of the virtual radius of the 'standard' beet.

scanned sugar beets, the following mean value of the ratio of D_2 and D_3 was found:

$$E(\frac{D_2}{D_3}) = 1.223$$

In other words: The sugar beets are (in average) somehow 'squeezed' or 'flat'.

However, the 'standard' beet will probably be a good model to use for localizing sugar beets placed in heaps. As soon as the position of a beet has been located, further refinement should by possible by attempts to match certain groups of distinctive beet shapes.

The localization of top slices appears to be possible. Figure 7.6 shows an example of beet with quite a lot of top, but also in a closer to average case like depicted in figure 7.7, the position of the top is distinctive.

All in all, there seems to be possibilities in abundance for exploiting geometrical properties in the search for sugar beets in 3D range data.

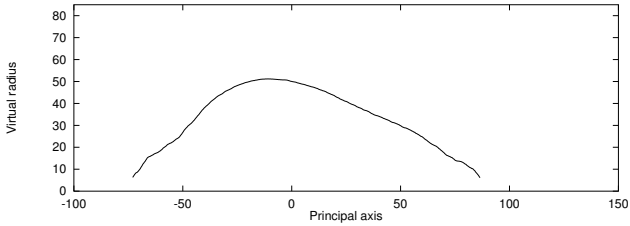


Figure 7.6: A beet with a high top percentage.

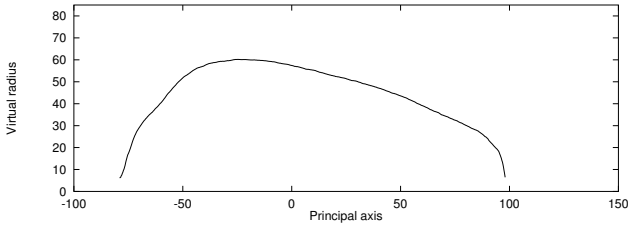


Figure 7.7: A beet with a 'normal' percentage.

Conclusion

In this thesis the concept of scanning 3D objects from triangulation by the means of CCD-cameras and a sheet of laser light has been throughly investigated.

A laser scanner has been constructed from scratch using off-the-shelf components and a few customized parts. This has been a time consuming but very interesting process revealing a lot of features as well as problems attributed to laser scanners. The results of this thesis are not only due to efforts of the mind, but have been accomplished from a great deal of hands-on activity too. The project has been comprehensive including many different aspects of engineering. That includes hardware as well as software problems.

The final product, the laser scanner, has become a versatile and functional apparatus. It is well performing. Estimates have been conducted on the accuracy of the scanner, and the result were very fine for equipment of this low cost.

Accuracy has been a main issue constructing the laser scanner, and demands to accuracy have been included in every part of the process of the construction.

Two different methods for camera calibration have been adopted as part of the system. The Tsai algorithm proved better performance in the process of calibration, but the direct linear transformation was not significantly inferior to the Tsai algorithm when utilized in the process of scanning. It is though conceivable, that performance differences will come clear when larger objects are scanned.

Different methods for the estimation of the axis of rotation have been investigated. The simplest one was found to yield the most robust results.

An extra CCD-camera was included in the system to improve performance and avoid some the problems and error sources inherent to the scanner system. Including another sensor raised the demand on software implementation, but proved beneficial. A great deal of occlusion problems were avoided and data redundancy improved the quality of range data.

It has been shown, that the scanner indeed is capable of producing high quality range data, and the scanner has been deployed for the task of scanning a number of sugar beets to prove its functionality. The results have been good, and the data has been the basis for some preliminary investigations on 3D measurements of sugar beets.

A number of shape features and dimensional properties have been calculated for the scanned sugar beets. The results are promising and further investigation is bound to reveal interesting conclusions. Many features are distinctive in the plots of the virtual radius, but it probably will be rewarding to do some calculations on the surface properties as well - in particular with respect to the problem of locating the top slice of the beet. Sadly, due to the limited time it has not been possible to go further in this direction.

For future work some improvements to the scanner system are suggested. First of all, it would be a good idea to deploy some fine-mechanics for the alignment of the laser beam and the calibration plane. It would also be interesting to do a high-speed implementation exploiting some of the functionalities of the hardware. For better results in daylight, a band-pass filter should be added to the sensor system.

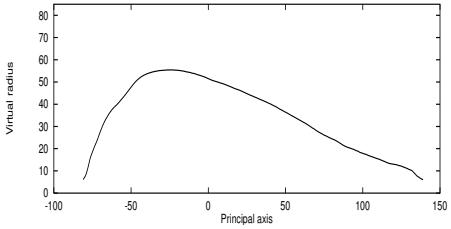
Appendix A

Sugar Beets - Shape Properties

Note: 3D-renderings not to scale.

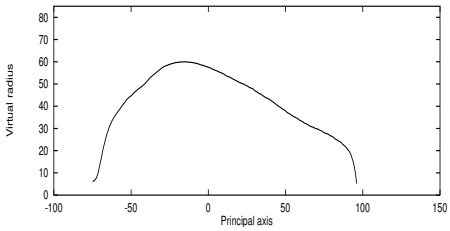
Figure A.1: Beet 1

$\mathbf{D}_1$: 0.623
$\mathbf{D}_2$: 0.208
$\mathbf{D}_3$: 0.169
$\mathbf{l}_1$: 222mm
Volume: 1015ml



Shape properties and virtual radius, Beet 1.

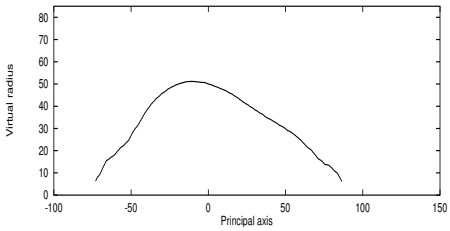
Figure A.2: Beet 2



$\mathbf{D}_1$: 0.515
$\mathbf{D}_2$: 0.260
$\mathbf{D}_3$: 0.226
$\mathbf{l}_1$: 174mm
Volume: 1068ml

Shape properties and virtual radius, Beet 2.

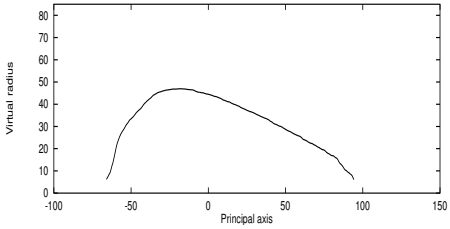
Figure A.3: Beet 3



$\mathbf{D}_1$: 0.510
$\mathbf{D}_2$: 0.259
$\mathbf{D}_3$: 0.230
$\mathbf{l}_1$: 164mm
Volume: 661ml

Shape properties and virtual radius, Beet 3.

Figure A.4: Beet 4



$\mathbf{D}_1$: 0.589
$\mathbf{D}_2$: 0.232
$\mathbf{D}_3$: 0.179
$\mathbf{l}_1$: 169mm
Volume: 613ml

Shape properties and virtual radius, Beet 4.

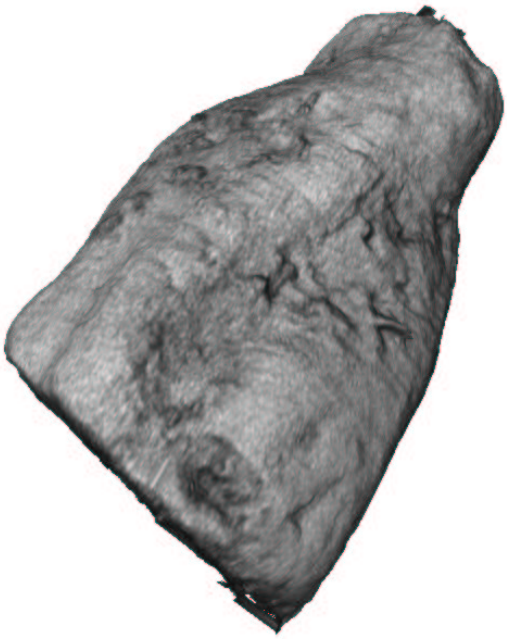
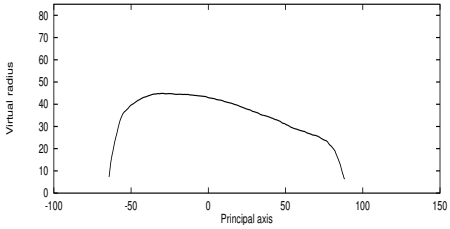


Figure A.5: Beet 5

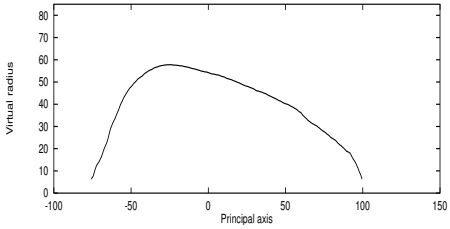


$\mathbf{D}_1$: 0.621
$\mathbf{D}_2$: 0.208
$\mathbf{D}_3$: 0.171
$\mathbf{l}_1$: 155mm
Volume: 643ml

Shape properties and virtual radius, Beet 5.

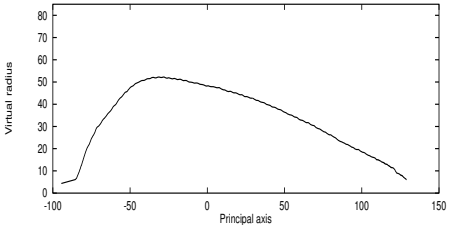
Figure A.6: Beet 6

$\mathbf{D}_1$: 0.538
$\mathbf{D}_2$: 0.264
$\mathbf{D}_3$: 0.198
$\mathbf{l}_1$: 176mm
Volume: 1051ml



Shape properties and virtual radius, Beet 6.

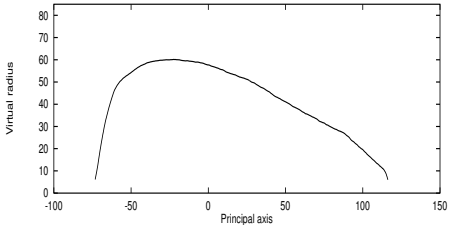
Figure A.7: Beet 7



$\mathbf{D}_1$: 0.660
$\mathbf{D}_2$: 0.192
$\mathbf{D}_3$: 0.149
$\mathbf{l}_1$: 223mm
Volume: 959ml

Shape properties and virtual radius, Beet 7.

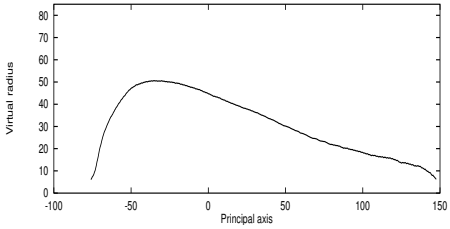
Figure A.8: Beet 8



$\mathbf{D}_1$: 0.558
$\mathbf{D}_2$: 0.261
$\mathbf{D}_3$: 0.181
$\mathbf{l}_1$: 176mm
Volume: 1241ml

Shape properties and virtual radius, Beet 8.

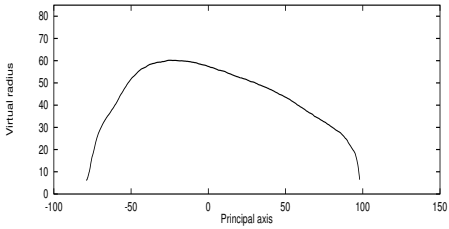
Figure A.9: Beet 9



$\mathbf{D}_1$: 0.695
$\mathbf{D}_2$: 0.173
$\mathbf{D}_3$: 0.132
$\mathbf{l}_1$: 227mm
Volume: 811ml

Shape properties and virtual radius, Beet 9.

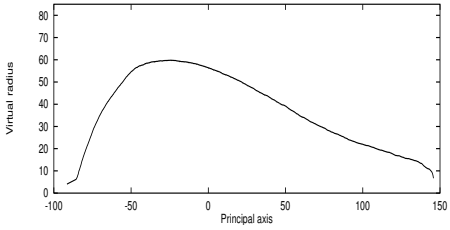
Figure A.10: Beet 10



$\mathbf{D}_1$: 0.554
$\mathbf{D}_2$: 0.240
$\mathbf{D}_3$: 0.205
$\mathbf{l}_1$: 169mm
Volume: 1229ml

Shape properties and virtual radius, Beet 10.

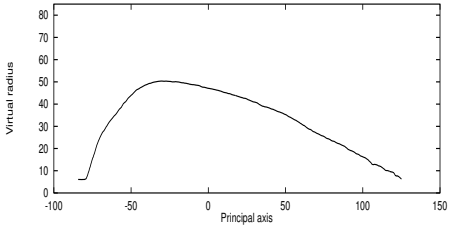
Figure A.11: Beet 11



$\mathbf{D}_1$: 0.615
$\mathbf{D}_2$: 0.210
$\mathbf{D}_3$: 0.175
$\mathbf{l}_1$: 235mm
Volume: 1250ml

Shape properties and virtual radius, Beet 11.

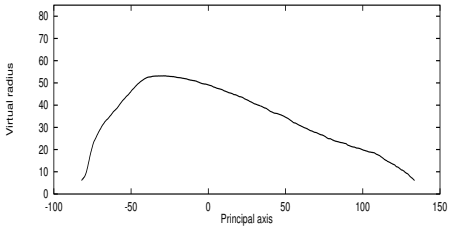
Figure A.12: Beet 12



$\mathbf{D}_1$: 0.649
$\mathbf{D}_2$: 0.195
$\mathbf{D}_3$: 0.156
$\mathbf{l}_1$: 213mm
Volume: 858ml

Shape properties and virtual radius, Beet 12.

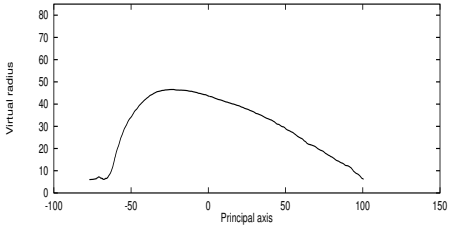
Figure A.13: Beet 13



$\mathbf{D}_1$: 0.654
$\mathbf{D}_2$: 0.192
$\mathbf{D}_3$: 0.153
$\mathbf{l}_1$: 220mm
Volume: 941ml

Shape properties and virtual radius, Beet 13.

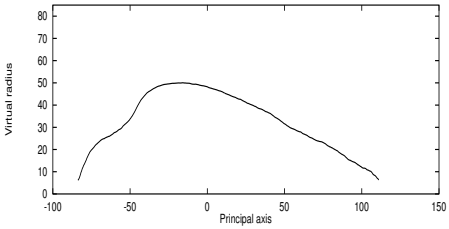
Figure A.14: Beet 14



$\mathbf{D}_1$: 0.593
$\mathbf{D}_2$: 0.240
$\mathbf{D}_3$: 0.168
$\mathbf{l}_1$: 180mm
Volume: 614ml

Shape properties and virtual radius, Beet 14.

Figure A.15: Beet 15

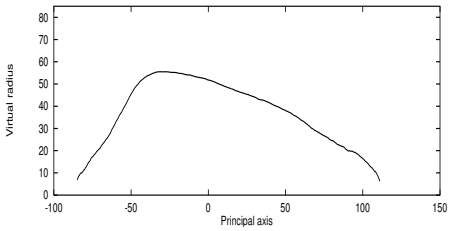


$\mathbf{D}_1$: 0.617
$\mathbf{D}_2$: 0.210
$\mathbf{D}_3$: 0.173
$\mathbf{l}_1$: 197mm
Volume: 766ml

Shape properties and virtual radius, Beet 15.

Figure A.16: Beet 17

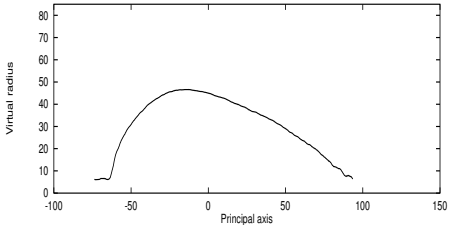
$\mathbf{D}_1$: 0.581
$\mathbf{D}_2$: 0.228
$\mathbf{D}_3$: 0.191
$\mathbf{l}_1$: 196mm
Volume: 982ml



Shape properties and virtual radius, Beet 17.

Figure A.17: Beet 18

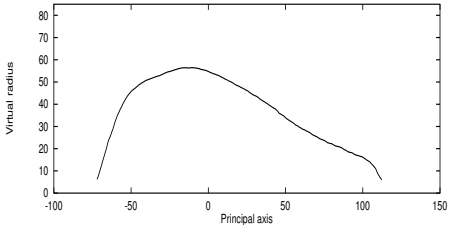
$\mathbf{D}_1$: 0.585
$\mathbf{D}_2$: 0.220
$\mathbf{D}_3$: 0.195
$\mathbf{l}_1$: 174mm
Volume: 594ml



Shape properties and virtual radius, Beet 18.

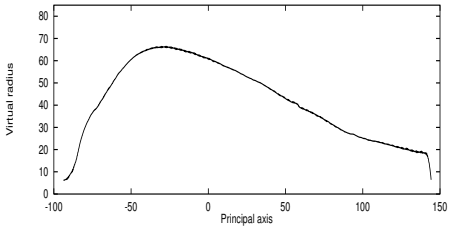
Figure A.18: Beet 19

$\mathbf{D}_1$: 0.561
$\mathbf{D}_2$: 0.250
$\mathbf{D}_3$: 0.188
$\mathbf{l}_1$: 185mm
Volume: 951ml



Shape properties and virtual radius, Beet 19.

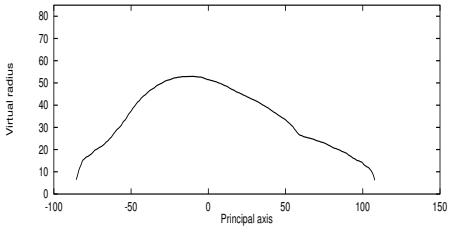
Figure A.19: Beet 21



$\mathbf{D}_1$: 0.598
$\mathbf{D}_2$: 0.217
$\mathbf{D}_3$: 0.185
$\mathbf{l}_1$: 238mm
Volume: 1554ml

Shape properties and virtual radius, Beet 21.
Note: Beet dimension exceeds view of scanner.

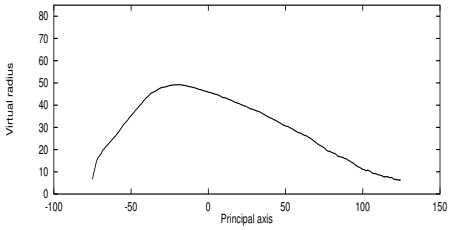
Figure A.20: Beet 22



$\mathbf{D}_1$: 0.600
$\mathbf{D}_2$: 0.226
$\mathbf{D}_3$: 0.174
$\mathbf{l}_1$: 194mm
Volume: 839ml

Shape properties and virtual radius, Beet 22.

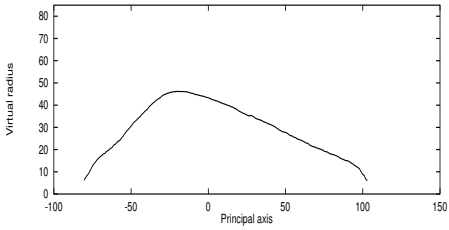
Figure A.21: Beet 23



$\mathbf{D}_1$: 0.624
$\mathbf{D}_2$: 0.200
$\mathbf{D}_3$: 0.176
$\mathbf{l}_1$: 204mm
Volume: 705ml

Shape properties and virtual radius, Beet 23.

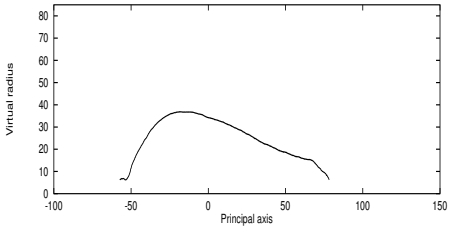
Figure A.22: Beet 24



$\mathbf{D}_1$: 0.644
$\mathbf{D}_2$: 0.188
$\mathbf{D}_3$: 0.168
$\mathbf{l}_1$: 184mm
Volume: 591ml

Shape properties and virtual radius, Beet 24.

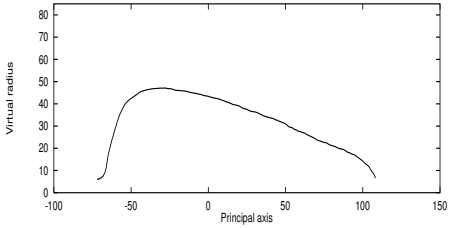
Figure A.23: Beet 25



$\mathbf{D}_1$: 0.633
$\mathbf{D}_2$: 0.201
$\mathbf{D}_3$: 0.166
$\mathbf{l}_1$: 152mm
Volume: 301ml

Shape properties and virtual radius, Beet 25.

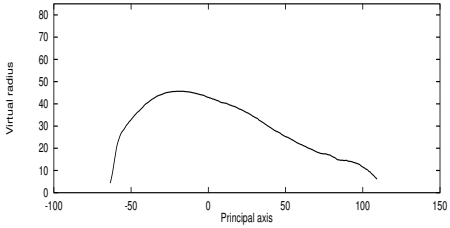
Figure A.24: Beet 26



$\mathbf{D}_1$: 0.651
$\mathbf{D}_2$: 0.200
$\mathbf{D}_3$: 0.149
$\mathbf{l}_1$: 183mm
Volume: 693ml

Shape properties and virtual radius, Beet 26.

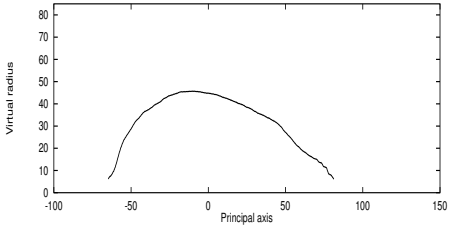
Figure A.25: Beet 27



$\mathbf{D}_1$: 0.639
$\mathbf{D}_2$: 0.188
$\mathbf{D}_3$: 0.174
$\mathbf{l}_1$: 164mm
Volume: 569ml

Shape properties and virtual radius, Beet 27.

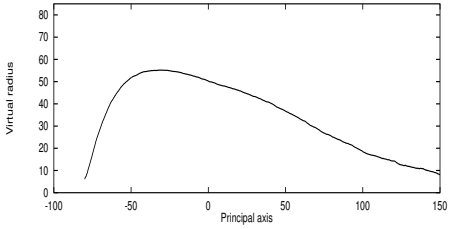
Figure A.26: Beet 28



$\mathbf{D}_1$: 0.531
$\mathbf{D}_2$: 0.256
$\mathbf{D}_3$: 0.213
$\mathbf{l}_1$: 149mm
Volume: 551ml

Shape properties and virtual radius, Beet 28.

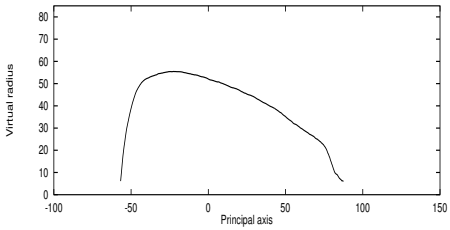
Figure A.27: Beet 29



$\mathbf{D}_1$: 0.643
$\mathbf{D}_2$: 0.193
$\mathbf{D}_3$: 0.164
$\mathbf{l}_1$: 234mm
Volume: 1042ml

Shape properties and virtual radius, Beet 29.

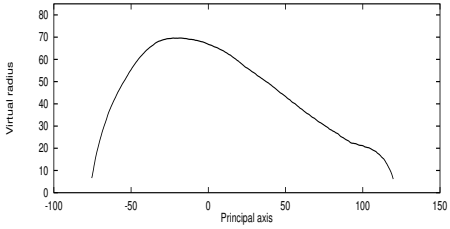
Figure A.28: Beet 30



$\mathbf{D}_1$: 0.479
$\mathbf{D}_2$: 0.269
$\mathbf{D}_3$: 0.252
$\mathbf{l}_1$: 145mm
Volume: 851ml

Shape properties and virtual radius, Beet 30.

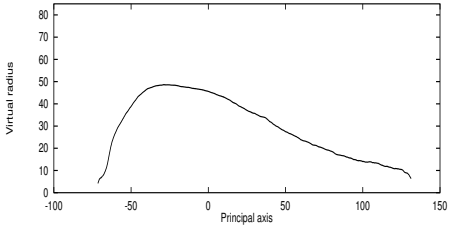
Figure A.29: Beet 31



$\mathbf{D}_1$: 0.451
$\mathbf{D}_2$: 0.291
$\mathbf{D}_3$: 0.257
$\mathbf{l}_1$: 183mm
Volume: 1497ml

Shape properties and virtual radius, Beet 31.

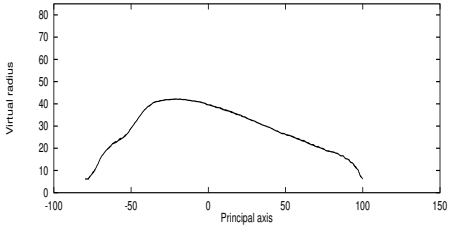
Figure A.30: Beet 32



$\mathbf{D}_1$: 0.646
$\mathbf{D}_2$: 0.182
$\mathbf{D}_3$: 0.171
$\mathbf{l}_1$: 190mm
Volume: 689ml

Shape properties and virtual radius, Beet 32.

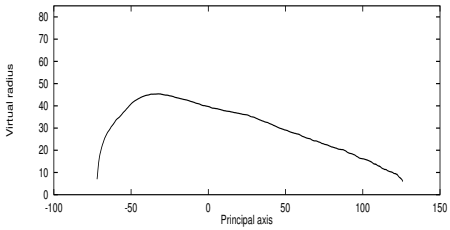
Figure A.31: Beet 33



$\mathbf{D}_1$: 0.686
$\mathbf{D}_2$: 0.170
$\mathbf{D}_3$: 0.144
$\mathbf{l}_1$: 181mm
Volume: 523ml

Shape properties and virtual radius, Beet 33.

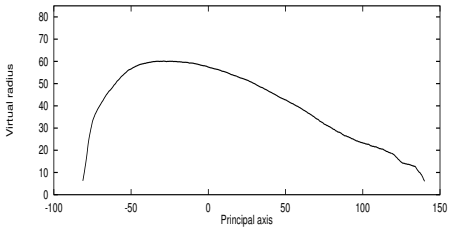
Figure A.32: Beet 35



$\mathbf{D}_1$: 0.722
$\mathbf{D}_2$: 0.149
$\mathbf{D}_3$: 0.129
$\mathbf{l}_1$: 199mm
Volume: 654ml

Shape properties and virtual radius, Beet 35.
Note: unsatisfactory removal of fixation device.

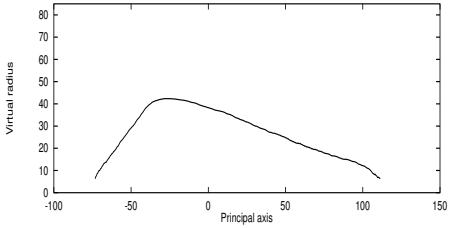
Figure A.33: Beet 36



$\mathbf{D}_1$: 0.602
$\mathbf{D}_2$: 0.219
$\mathbf{D}_3$: 0.179
$\mathbf{l}_1$: 222mm
Volume: 1358ml

Shape properties and virtual radius, Beet 36.

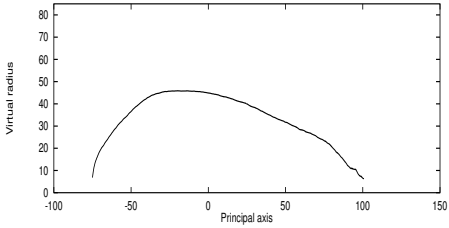
Figure A.34: Beet 37



$\mathbf{D}_1$: 0.694
$\mathbf{D}_2$: 0.171
$\mathbf{D}_3$: 0.134
$\mathbf{l}_1$: 194mm
Volume: 493ml

Shape properties and virtual radius, Beet 37.

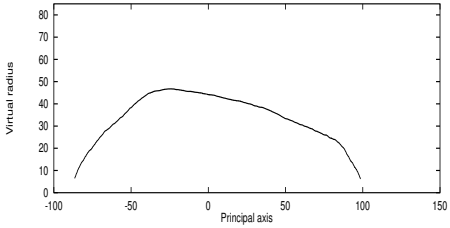
Figure A.35: Beet 38



$\mathbf{D}_1$: 0.635
$\mathbf{D}_2$: 0.206
$\mathbf{D}_3$: 0.159
$\mathbf{l}_1$: 177mm
Volume: 688ml

Shape properties and virtual radius, Beet 38.

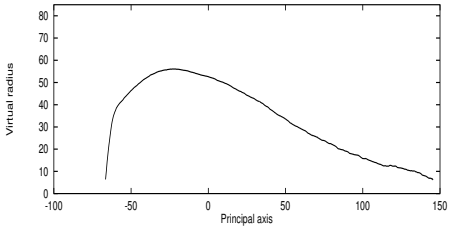
Figure A.36: Beet 39



$\mathbf{D}_1$: 0.667
$\mathbf{D}_2$: 0.179
$\mathbf{D}_3$: 0.154
$\mathbf{l}_1$: 186mm
Volume: 747ml

Shape properties and virtual radius, Beet 39.

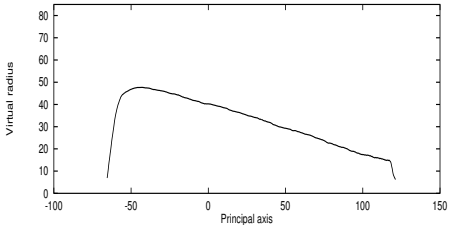
Figure A.37: Beet 40



$\mathbf{D}_1$: 0.585
$\mathbf{D}_2$: 0.220
$\mathbf{D}_3$: 0.194
$\mathbf{l}_1$: 214mm
Volume: 948ml

Shape properties and virtual radius, Beet 40.

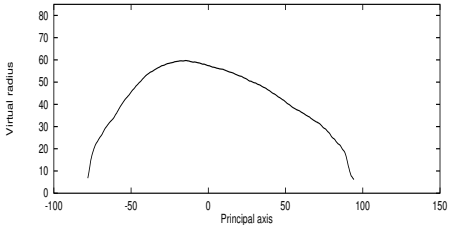
Figure A.38: Beet 41



$\mathbf{D}_1$: 0.704
$\mathbf{D}_2$: 0.155
$\mathbf{D}_3$: 0.142
$\mathbf{l}_1$: 187mm
Volume: 684ml

Shape properties and virtual radius, Beet 41.

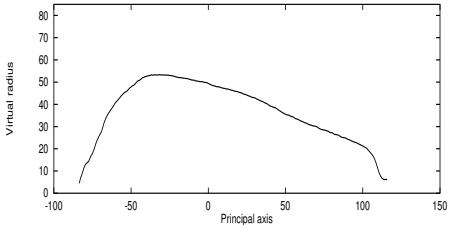
Figure A.39: Beet 42



$\mathbf{D}_1$: 0.515
$\mathbf{D}_2$: 0.262
$\mathbf{D}_3$: 0.222
$\mathbf{l}_1$: 173mm
Volume: 1125ml

Shape properties and virtual radius, Beet 42.

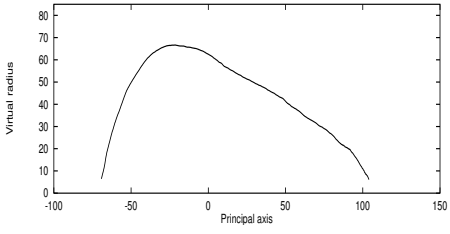
Figure A.40: Beet 43



$\mathbf{D}_1$: 0.636
$\mathbf{D}_2$: 0.200
$\mathbf{D}_3$: 0.164
$\mathbf{l}_1$: 195mm
Volume: 974ml

Shape properties and virtual radius, Beet 43.

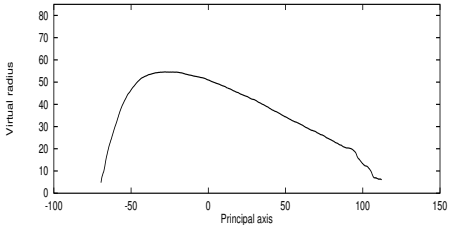
Figure A.41: Beet 44



$\mathbf{D}_1$: 0.510
$\mathbf{D}_2$: 0.268
$\mathbf{D}_3$: 0.221
$\mathbf{l}_1$: 174mm
Volume: 1256ml

Shape properties and virtual radius, Beet 44.

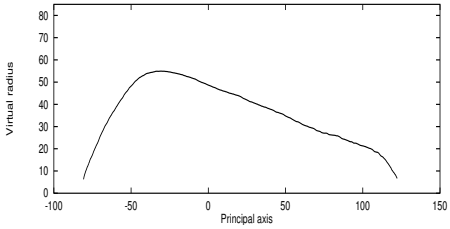
Figure A.42: Beet 45



$\mathbf{D}_1$: 0.560
$\mathbf{D}_2$: 0.261
$\mathbf{D}_3$: 0.179
$\mathbf{l}_1$: 172mm
Volume: 909ml

Shape properties and virtual radius, Beet 45.

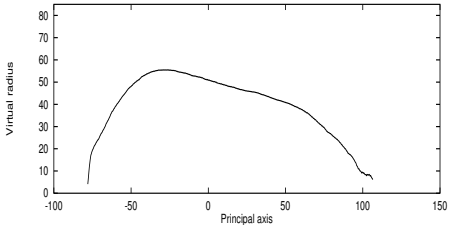
Figure A.43: Beet 46



$\mathbf{D}_1$: 0.640
$\mathbf{D}_2$: 0.202
$\mathbf{D}_3$: 0.159
$\mathbf{l}_1$: 203mm
Volume: 960ml

Shape properties and virtual radius, Beet 46.

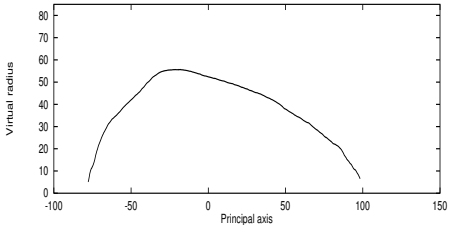
Figure A.44: Beet 47



$\mathbf{D}_1$: 0.583
$\mathbf{D}_2$: 0.223
$\mathbf{D}_3$: 0.194
$\mathbf{l}_1$: 175mm
Volume: 1028ml

Shape properties and virtual radius, Beet 47.

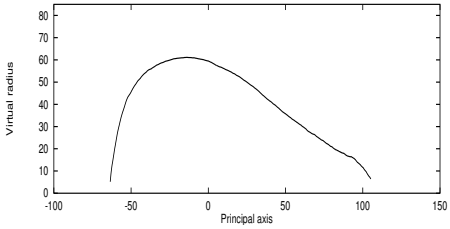
Figure A.45: Beet 48



$\mathbf{D}_1$: 0.547
$\mathbf{D}_2$: 0.257
$\mathbf{D}_3$: 0.196
$\mathbf{l}_1$: 175mm
Volume: 964ml

Shape properties and virtual radius, Beet 48.

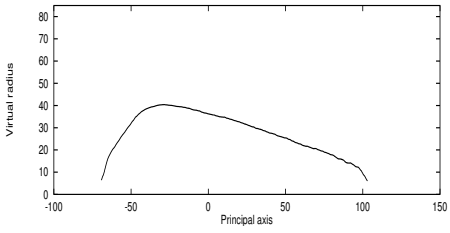
Figure A.46: Beet 49



$\mathbf{D}_1$: 0.467
$\mathbf{D}_2$: 0.310
$\mathbf{D}_3$: 0.222
$\mathbf{l}_1$: 162mm
Volume: 1053ml

Shape properties and virtual radius, Beet 49.

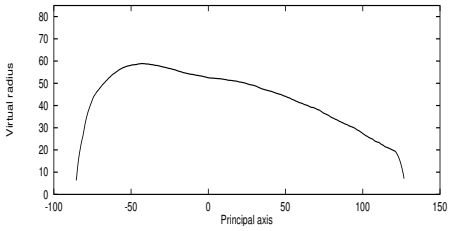
Figure A.47: Beet 50



$\mathbf{D}_1$: 0.714
$\mathbf{D}_2$: 0.161
$\mathbf{D}_3$: 0.125
$\mathbf{l}_1$: 174mm
Volume: 475ml

Shape properties and virtual radius, Beet 50.

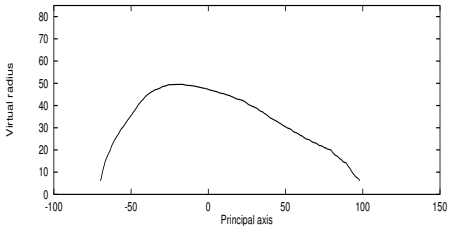
Figure A.48: Beet 51



$\mathbf{D}_1$: 0.654
$\mathbf{D}_2$: 0.194
$\mathbf{D}_3$: 0.152
$\mathbf{l}_1$: 214mm
Volume: 1379ml

Shape properties and virtual radius, Beet 51.
Note: Beet dimension exceeds view of scanner.

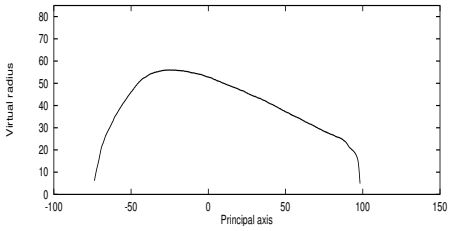
Figure A.49: Beet 52



$\mathbf{D}_1$: 0.576
$\mathbf{D}_2$: 0.220
$\mathbf{D}_3$: 0.204
$\mathbf{l}_1$: 167mm
Volume: 711ml

Shape properties and virtual radius, Beet 52.

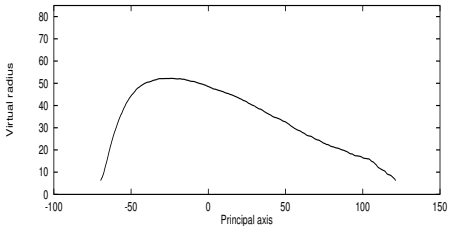
Figure A.50: Beet 53



$\mathbf{D}_1$: 0.581
$\mathbf{D}_2$: 0.242
$\mathbf{D}_3$: 0.176
$\mathbf{l}_1$: 174mm
Volume: 993ml

Shape properties and virtual radius, Beet 53.

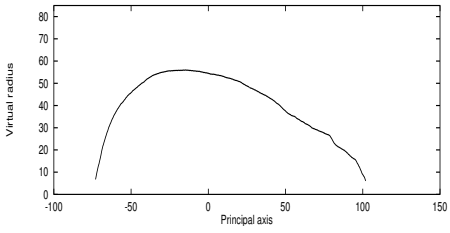
Figure A.51: Beet 54



$\mathbf{D}_1$: 0.604
$\mathbf{D}_2$: 0.211
$\mathbf{D}_3$: 0.184
$\mathbf{l}_1$: 192mm
Volume: 828ml

Shape properties and virtual radius, Beet 54.

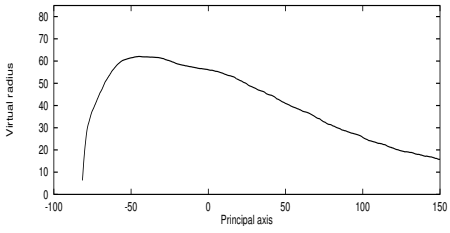
Figure A.52: Beet 55



$\mathbf{D}_1$: 0.538
$\mathbf{D}_2$: 0.250
$\mathbf{D}_3$: 0.212
$\mathbf{l}_1$: 176mm
Volume: 1020ml

Shape properties and virtual radius, Beet 55.

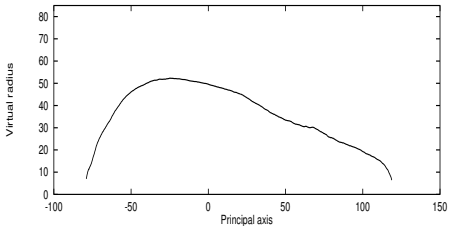
Figure A.53: Beet 56



$\mathbf{D}_1$: 0.650
$\mathbf{D}_2$: 0.185
$\mathbf{D}_3$: 0.165
$\mathbf{l}_1$: 232mm
Volume: 1436ml

Shape properties and virtual radius, Beet 56.

Figure A.54: Beet 58



$\mathbf{D}_1$: 0.646
$\mathbf{D}_2$: 0.187
$\mathbf{D}_3$: 0.167
$\mathbf{l}_1$: 198mm
Volume: 921ml

Shape properties and virtual radius, Beet 58.

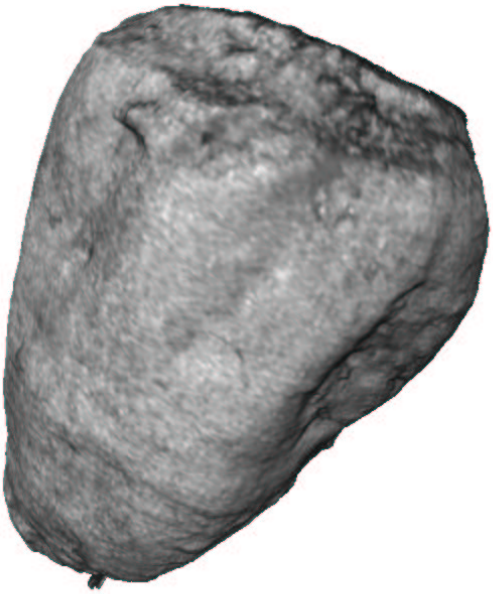
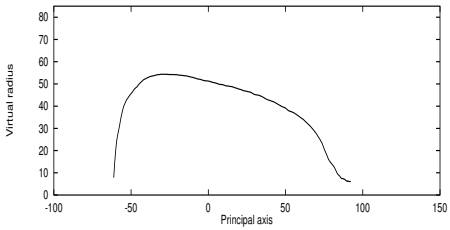


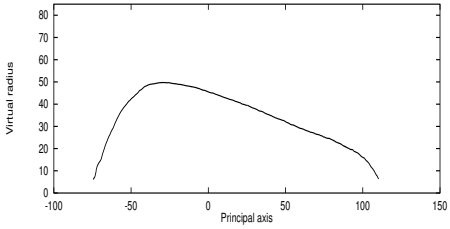
Figure A.55: Beet 59

$\mathbf{D}_1$: 0.504
$\mathbf{D}_2$: 0.262
$\mathbf{D}_3$: 0.234
$\mathbf{l}_1$: 158mm
Volume: 909ml



Shape properties and virtual radius, Beet 59.

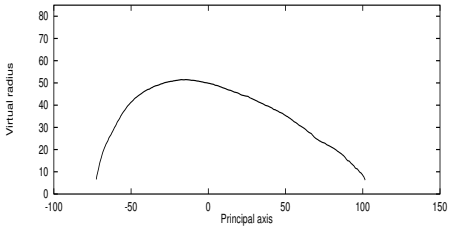
Figure A.56: Beet 60



$\mathbf{D}_1$: 0.641
$\mathbf{D}_2$: 0.206
$\mathbf{D}_3$: 0.153
$\mathbf{l}_1$: 186mm
Volume: 774ml

Shape properties and virtual radius, Beet 60.

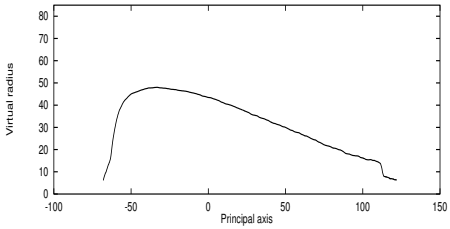
Figure A.57: Beet 61



$\mathbf{D}_1$: 0.607
$\mathbf{D}_2$: 0.225
$\mathbf{D}_3$: 0.168
$\mathbf{l}_1$: 174mm
Volume: 830ml

Shape properties and virtual radius, Beet 61.

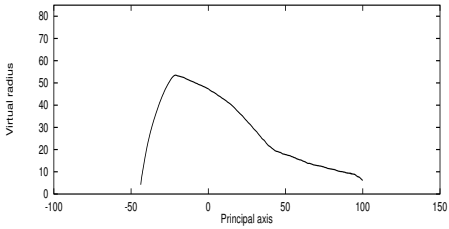
Figure A.58: Beet 62



$\mathbf{D}_1$: 0.667
$\mathbf{D}_2$: 0.200
$\mathbf{D}_3$: 0.134
$\mathbf{l}_1$: 181mm
Volume: 709ml

Shape properties and virtual radius, Beet 62.

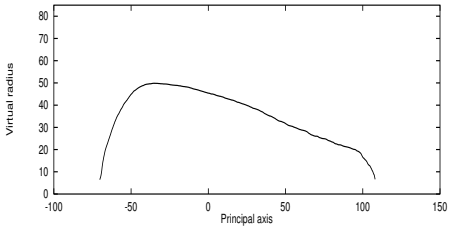
Figure A.59: Beet 63



$\mathbf{D}_1$: 0.381
$\mathbf{D}_2$: 0.340
$\mathbf{D}_3$: 0.279
$\mathbf{l}_1$: 148mm
Volume: 482ml

Shape properties and virtual radius, Beet 63.

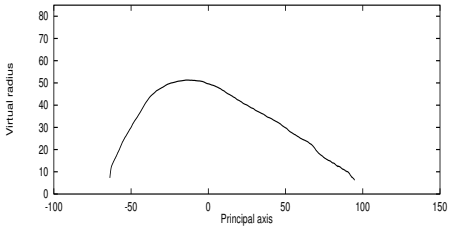
Figure A.60: Beet 64



$\mathbf{D}_1$: 0.630
$\mathbf{D}_2$: 0.219
$\mathbf{D}_3$: 0.151
$\mathbf{l}_1$: 179mm
Volume: 783ml

Shape properties and virtual radius, Beet 64.

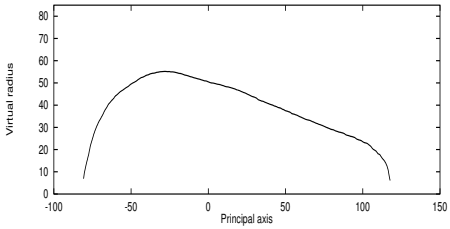
Figure A.61: Beet 65



$\mathbf{D}_1$: 0.514
$\mathbf{D}_2$: 0.264
$\mathbf{D}_3$: 0.222
$\mathbf{l}_1$: 159mm
Volume: 678ml

Shape properties and virtual radius, Beet 65.

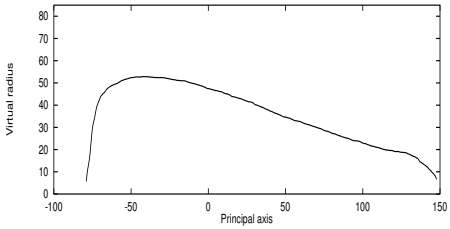
Figure A.62: Beet 66



$\mathbf{D}_1$: 0.667
$\mathbf{D}_2$: 0.176
$\mathbf{D}_3$: 0.157
$\mathbf{l}_1$: 199mm
Volume: 1067ml

Shape properties and virtual radius, Beet 66.

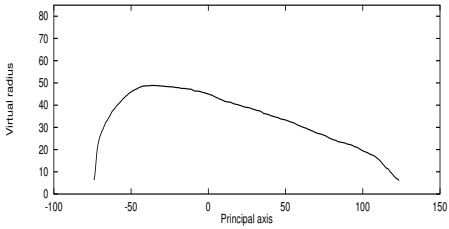
Figure A.63: Beet 67



$\mathbf{D}_1$: 0.706
$\mathbf{D}_2$: 0.158
$\mathbf{D}_3$: 0.135
$\mathbf{l}_1$: 226mm
Volume: 1046ml

Shape properties and virtual radius, Beet 67.

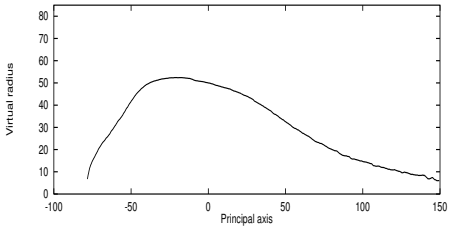
Figure A.64: Beet 68



$\mathbf{D}_1$: 0.690
$\mathbf{D}_2$: 0.165
$\mathbf{D}_3$: 0.144
$\mathbf{l}_1$: 197mm
Volume: 821ml

Shape properties and virtual radius, Beet 68.

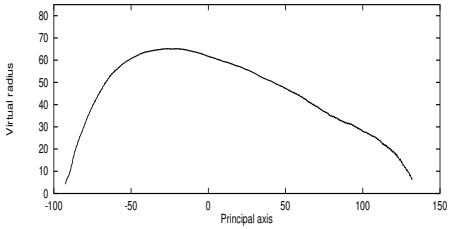
Figure A.65: Beet 69



$\mathbf{D}_1$: 0.617
$\mathbf{D}_2$: 0.224
$\mathbf{D}_3$: 0.160
$\mathbf{l}_1$: 232mm
Volume: 863ml

Shape properties and virtual radius, Beet 69.

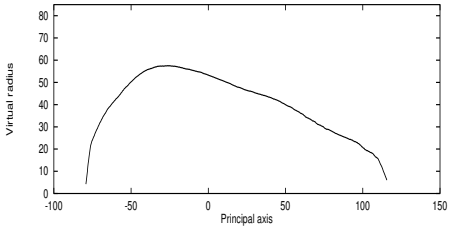
Figure A.66: Beet 70



$\mathbf{D}_1$: 0.585
$\mathbf{D}_2$: 0.228
$\mathbf{D}_3$: 0.187
$\mathbf{l}_1$: 218mm
Volume: 1644ml

Shape properties and virtual radius, Beet 70.

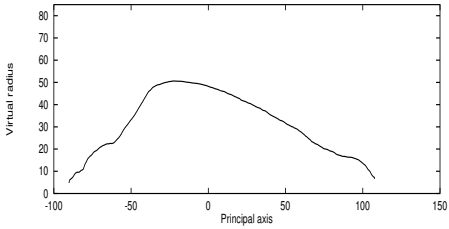
Figure A.67: Beet 71



$\mathbf{D}_1$: 0.612
$\mathbf{D}_2$: 0.210
$\mathbf{D}_3$: 0.178
$\mathbf{l}_1$: 190mm
Volume: 1121ml

Shape properties and virtual radius, Beet 71.

Figure A.68: Beet 72



$\mathbf{D}_1$: 0.601
$\mathbf{D}_2$: 0.211
$\mathbf{D}_3$: 0.187
$\mathbf{l}_1$: 192mm
Volume: 751ml

Shape properties and virtual radius, Beet 72.

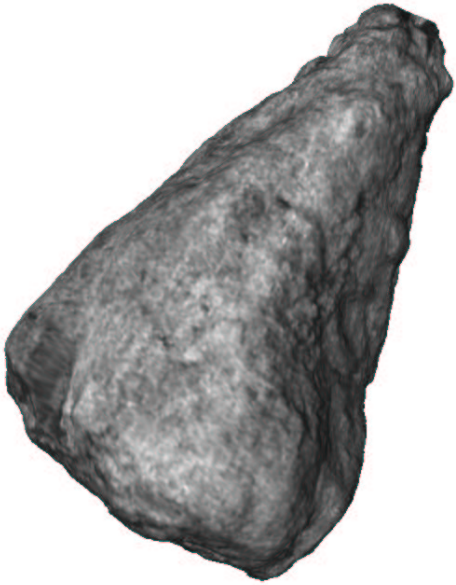
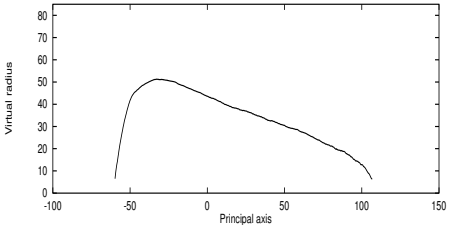


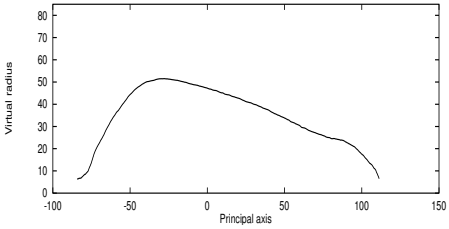
Figure A.69: Beet 73



$\mathbf{D}_1$: 0.603
$\mathbf{D}_2$: 0.209
$\mathbf{D}_3$: 0.188
$\mathbf{l}_1$: 168mm
Volume: 692ml

Shape properties and virtual radius, Beet 73.

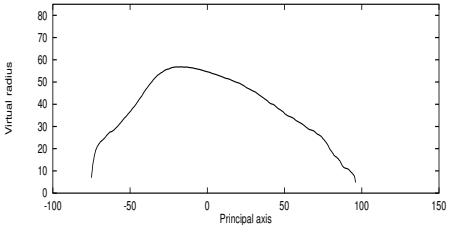
Figure A.70: Beet 74



$\mathbf{D}_1$: 0.641
$\mathbf{D}_2$: 0.180
$\mathbf{D}_3$: 0.178
$\mathbf{l}_1$: 197mm
Volume: 855ml

Shape properties and virtual radius, Beet 74.

Figure A.71: Beet 75



$\mathbf{D}_1$: 0.504
$\mathbf{D}_2$: 0.282
$\mathbf{D}_3$: 0.214
$\mathbf{l}_1$: 174mm
Volume: 930ml

Shape properties and virtual radius, Beet 75.

Appendix B

Hardware

B.1 Electrical Diagram

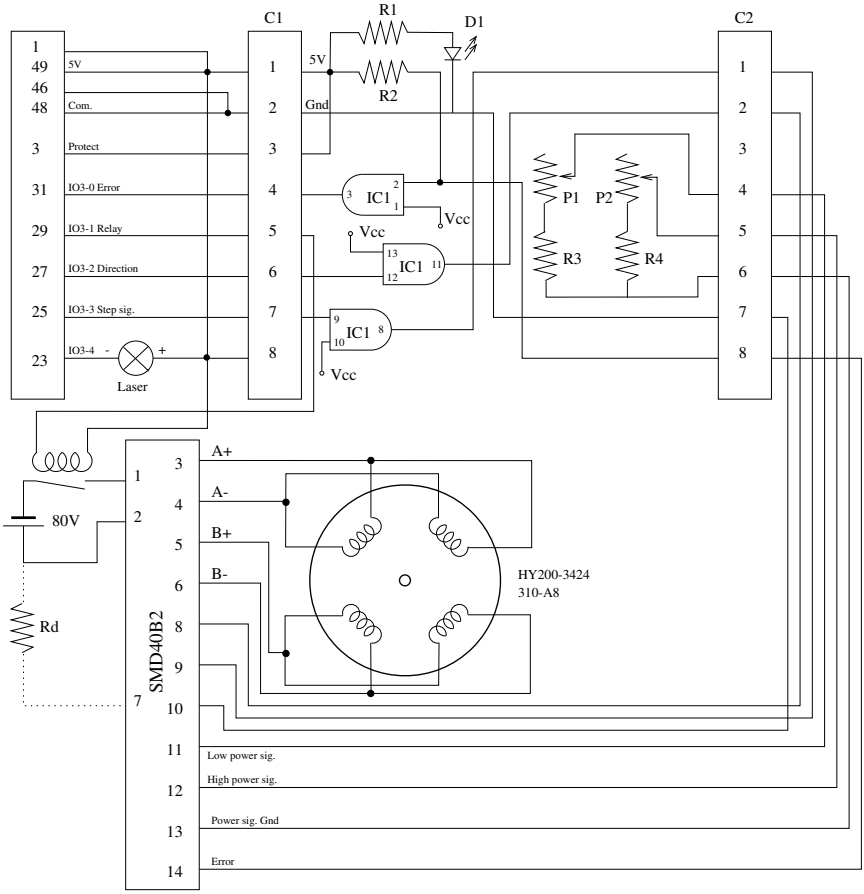


Figure B.1: Wiring diagram.

B.2 Hardware Item List

C_1, C_2	8-pin connectors
R_1	150Ω
R_2	1kΩ
R_3	4.7kΩ
R_4	47kΩ
R_d	60Ω, 70W
P_1	47kΩ
P_2	470kΩ
D_1	LED
IC_1	HC7408 (4 × AND)

SM87.1.18.j3.248 Zebotronic step motor
SMD40B2 Step motor controller
80V power supply
10A relay

Appendix C

Code

All code written in C.

C.1 Main Programs

Name

rot

Synopsis

rot

Description

Do laser scan.

Files

Configuration files:

rotscan.cfg - Optional, using defaults if file is not present.
Must contain all 5 lines.
Line 1: Number of steps per scan (integer).
Line 2: Speed (max 500) (double).
Line 3: Image threshold for laser detection (unsigned char).
Line 4: High speed (double).

Line 5: Intensity mapping: 0 = none, 1 = laser reflectance, 2 = ambient light (integer).

From calibration routine:

cam0Mmat.dat - required.
cam1Mmat.dat - required.
cam0cpc.dat - required.
cam1cpc.dat - required.
rotax.df - required.

Ouput:

Scan0.cd - Coordinate and intensity data, camera 0.
Scan1.cd - Coordinate and intensity data, camera 1.

Defaults

Number of steps per scan: 360 + padding = 362.
Image threshold for laser detection : 80.
Speed: 150.
Intensity mapping: Laser reflectance.

Name

compcal

Synopsis

compcal

Description

Calibrates laser scanner. Calibration plane must be alligned with laser sheet prior to call.

Files

Output:

cam0Mmat.dat
cam1Mmat.dat
cam0cpc.dat
cam1cpc.dat
rotax.df

Name

interpol

Synopsis

interpol < cdfile1 >< cdfile2 >

Description

interpol merges scan data from the two data files *cdfile1* and *cdfile2*.

Files

Output:

merge.cd: Merged data file.

Name

2iv

Synopsis

2iv [-f] < cdfile > < ivfile >

Description

Convert scan data in file *cdfile* to Open Inventor file *ivfile*.
Option -f: Filter void data.

Fileformat for scan data files

Binary format:

1 integer format: 0 = No texture mapping. 1 = Texture mapping.
1 integer nr: Number of scan lines.
1 integer nc: Number of points per scan line (usually 768).
3·nr·nc floats: (x,y,z) coordinates.
nr·nc unsigned char: Texture map (only if format = 1).

C.2 Header File, I/O Routines

rot_util.h

```
#define GEAR 10.5
#define MINISTEPS 10
#define MOTORSTEPS 200


#define steps2degs(X) 360*X/(MOTORSTEPS*MINISTEPS*GEAR)
#define degs2steps(X) X*MOTORSTEPS*MINISTEPS*GEAR/360
#define scans2stepsize(X) MOTORSTEPS*MINISTEPS*GEAR/X


unsigned short getIPNumber();
```

This function returns the IPnn number – the CPU board model number of the current system. For example in an Onyx it returns 21 (IP21)

```
int rotMoveByStep_64(int steps, int min_per);
```

Clock steps steps on the step signal output. The frequency is determined by min_per: Every step signal will have a duration of minimum min_per cycle counter clocks (21 picosecs on an Onyx system).

```
int rotate_fin(void);
```

Shutdown scanner

int rotate_init(void);

Initialize scanner system. Must be done to utilize system

double rotFreq2Ticks(double freq);

Return value: Number of tick corresponding to clock frequency freq.

double rotMoveByDeg(double deg, double speed);

Rotate table deg degrees at speed. Return value: Degrees actually rotated.

int rotWaitTicks64(__uint64_t ticks);

Wait ticks cyckle counter ticks (21 picosecs on Onyx systems).

int rotLaserOn();

Turn laser on

int rotLaserOff();

Switch laser off

Bibliography

[1] Laser diode structured light products. Product pamphlet, May 1995.

[2] Acromag, Wixon,Michigan. *Series 9510-X VME Multi-Function Card.*

[3] O. Brande-Lavridsen. *Fotogrammetri.* Lab. for Fotogrammetri og Landmåling, Ålborg, 8'th edition, 1990.

[4] Knut Conradsen. *En introduktion til Statistik, 1A,1B,2A & 2B.* Imsor, Lyngby, 4'th edition, 1984.

[5] Brian Curless and Marc Levoy. Better optical triangulation through spacetime analysis. In *5th International Conference on Computer Vision (ICCV95)*, pages 987–994, June 1995.

[6] Agamemnon Despopoulos and Stefan Silbernagl. *Color Atlas of Physiology.* Thieme, Stuttgart, 3'rd edition, 1986.

[7] Keld Dueholm. *Noter til Indledende billedbehandling.* Institut for Landmåling og Fotogrammetri, Lyngby, 1995.

[8] B.K.P. Horn. *Robot Vision.* The MIT Electrical Engineering and Computer Science Series. The MIT Press, Cambridge, Massachusetts, 1986.

[9] Ole Jacobi. *Fotogrammetri - Matematisk grundlag.* Institut for Landmåling og Fotogrammetri, Lyngby, 1984.

[10] R.A. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-5(2):122–139, March 1983.

[11] JVL Industri Elektronik A/S, Birkerod. *Stepmotor ministep driver*, April 1995.

[12] Karl Kraus. *Photogrammetry*, volume 1. Dümmler, Bonn, fourth edition, 1993.

[13] R.K. Lenz and R.Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-d machine vision metrol-

ogy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713–720, September 1988.

[14] Amer.Soc. of Photogrammetry. *Manual of Photogrammetry*. Amer.Soc. of Photogrammetry, 4'th edition, 1980.

[15] Robert J. Schalkoff. *Digital Image Processing and Computer Vision*. John Wiley & Sons, Inc, N.Y., 1989.

[16] Torben Skettrup. *Laserteknik I*. Polyteknisk Forlag, Lyngby, 4'th edition, 1985.

[17] Tecnoingranaggi Riduttori, Calderara di Reno. *Riduttori epicicloidali di precisione*. Product specifications.

[18] Roger Y. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 364–374, Miami Beach, FL, 1986.

[19] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, August 1987.

[20] Zebotronics, Munich. *Zebotrocnics Stepping Motors*, October 1994.