3D visual rapid robot programming system

Diego Adolfo Rodriguez Cantor
University of Wollongong

3D VISUAL RAPID ROBOT PROGRAMMING SYSTEM

A thesis submitted in fulfilment of the requirements for the award of the degree

Doctor of Philosophy (PhD)

From

UNIVERSITY OF WOLLONGONG

By

DIEGO ADOLFO RODRIGUEZ CANTOR

Master of Engineering Practice in Mechatronics

Electronics Engineer

FACULTY OF ENGINEERING

2008

CERTIFICATION

I, Diego Adolfo Rodríguez Cantor, declare that this thesis, submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

_____
Diego Adolfo Rodríguez Cantor
27 January 2008

ACKNOLEDGEMENTS

I wish to acknowledge the help and support provided by Professor John Norrish who has always had confidence in me. I also have to thank the advice given by Doctor Alex Nicholson throughout this work.

I can not forget to thank my marvellous wife Catherine who always gave me support and encouragement to accomplish this stage of my professional life.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS AND NOTATION

mm	millimetres
m	metres
mW	milliWatts
O_c	Projection focal point
x_f, y_f, z_f	Focal plane coordinate frame axes
x_i, y_i, z_i	Image plane coordinate frame axes
x_w, y_w, z_w	World frame axes
x_c, y_c	2 dimensional image camera frame axes
f	Focal distance
fx	Focal distance in pixels (using the scale in x)
fy	Focal distance in pixels (using the scale in y)
α_x	Image scale in x (pixels/mm)
α_y	Image scale in y (pixels/mm)
$\mathbf{R}$	Rotation matrix
$\mathbf{T}$	Translation matrix
•	Dot product
$\otimes$	Cross product
$\|\ \|$	Norm

ABSTRACT

Robotic welding may be used to enhance quality and improve operating conditions in manufacturing and repair situations. However, it may be difficult to implement robot automation for some applications which involve non-repetitive tasks such as in repair and reclamation work. The present work was aimed to address some of the robotic automation difficulties that arise in such situations where it is necessary to rapidly generate a robot program for every different task, in particular for weld repair.

A technique for rapidly generating off-line robot programs for weld repair was developed in this work. The system proposed uses a single camera in order to obtain information about the object to be welded. To perform the task, it is not necessary to know the geometry of the system "*a priori*" although smoothness of the overall shape of the object is assumed. Artificial feature points are used with the aim of being able to obtain the characteristics of the shape of the object in 3 dimensions.

A software application was developed which provides the system with an easy to use interface, sequence control and data processing which in combination with a vision system and a tactile technique allows weld repair jobs to be performed on complex curved surfaces.

A review of two camera calibration techniques was carried out as well as a review of commercial devices and techniques used to map 3D surfaces. A modification to the

planes camera calibration method was suggested and tested. Two methods to estimate 3D coordinates were review and tested, additionally a combination of these two methods was also tested. Results showing the errors obtained with the various combinations of camera calibration methods and 3D estimation methods were obtained. A technique to “unfold” a three dimensional surface (i.e. to create a two dimensional representation of a three dimensional surface) was proposed and used as an essential part of the image data processing. Tests were performed to evaluate the system and encouraging results were obtained.

1. INTRODUCTION

1.1 OBJECTIVE

The objective of the current work was to develop a technique to generate robot programs for different welding reclamation tasks involving complex 3D surfaces. In repair situations every application is different and manual robot programming is not viable. To offer a useable alternative the automatic generation of robot programs must be achieved reliably in a very short time. To accomplish this, a profile measuring and robot programming system and a simple, user friendly interface needed to be developed. The orientation and position of the work-piece should be not a constraint as long as it is visible by a camera; however there were expected to be limitations in distance and maximum angles over which acceptable results could be achieved. The final methodology, performance and prototype system were intended to be the outcome of this investigation.

The system was initially intended to perform welding tasks but other possible applications could be considered.

1.2 BACKGROUND

Welding repair jobs are non repetitive tasks and must often be performed on site, where conditions may pose additional risks to the welder. Because of this, it is desirable to automate such tasks but automation is generally regarded as

difficult or expensive or both. Manually generating a robot program to perform a weld repair task is time consuming and impractical. Therefore, in order to use robots to automate welding repair tasks a means of generating robot programs in a rapid way is required.

Several different off-line-programming techniques have been investigated and developed. Some involve CAD modelling [1-3]; this implies that, although the user may specify a given job, the geometry of the work-piece must be known *a priori*. The use of CAD models usually increases the cost of the system considerably [4]. There are other techniques that are based on pre-programmed small tasks which can be linked to each other by means of a user interface program. Other techniques involve machine vision but are limited to two dimensional (2D) surfaces, where the work-piece is in a plane at a known fixed distance [5]. Other vision based techniques are capable of acquiring 3D data but are often limited to a very specific job or restricted to simple surfaces (typically planes) [6].

Work by Nicholson [7] addressed some of these issues by allowing the user to select arbitrary target areas on a 2D work-piece and then using a “Touch Sensing” technique to refine the calculations.

The use of Machine vision has been used for several applications and has proved to be a very powerful tool in automation of tasks such as inspecting, assembling, welding and mapping [8]. Currently, it is possible to obtain 3D information using accurate devices such as 3D scanners which usually employ

laser technology. These systems can be used to obtain information of the environment geometry for generating robot programs. Although these devices are quite accurate, their price is restrictive for many small and medium businesses. Consequently, there is a need for a flexible cost-effective technique for rapid generation of robot programs for small batches or repair situations.

In order to explore alternative approaches it is necessary to review the techniques used for machine vision, including the optical systems and off-line programming methods.

2. LITERATURE REVIEW

2.1 REPAIR WELDING

Many centuries ago broken metal tools were repaired by heating and hammering to forge weld them. These may be considered the first repairs by welding [9]. Nowadays, several different techniques exist to execute a repair welding depending on the part or object to be repaired.

In general, a repair is performed when the replacement of the damaged part is expensive or part of a larger structure, to avoid loss of production or when replacements are not readily available [10]. Gregory [9] divides the defects that need repair by welding in two categories: fabrication defects and service failures. He considers service failures as "*cracks caused mainly by fatigue, brittle fracture, stress corrosion or creep".* The repair of service failures may not be easy due to factors such as restricted access to the part or a hazardous environment [9]. These factors can pose a risk to the welder in addition to the normal hazards involved in a welding procedure (electrical shock, arc radiation, air contamination, fire and explosion, compressed gases, etc [11]) and ergonomic issues such as the welder posture and the weight of the welding equipment.

To avoid exposing technicians to the risks involved in many weld repair jobs, to compensate for the shortage of skilled labour and to achieve good quality, the

automation of welding has gained importance [12]. Tung et al. [12] for example proposed an automated robotic system for weld repair of cracks using SMAW. The “Hephaestos ESPRIT 6042” project is another example of robot automation for weld repair but specially developed for the ship industry [13]. Nicholson [7] proposed and validated a technique for robotic weld repair of turbine runners. Sun et al. [14] proposed, simulated and experimented with a robotic welding system that could be used for repair jobs where an operator manipulates the robot remotely with assistance of sensors and software and control components.

2.2 ROBOTICS

The first industrial robot, the “Unimate” manufactured by Unimation Inc., was introduced in 1961. It was capable of storing 200 data points and to handle payloads up to 25 pounds at full speed [15]. Robots have evolved significantly since then and, according to Sprovieri [15], “*they have become smarter, faster, stronger, smaller and more accurate*”. An example of the great evolution of industrial robots is the two arm robot “DIA10” developed by MOTOMAN INC. [16]. This robots looks like a human torso which allows it to perform operations that previously only people could do [17].

Angelo [18] defines robots as: “*Smart machines with manipulators that can be programmed to do a variety of manual or human labour tasks*”. In the most typical configuration an industrial robot is composed of a manipulator, an end effector, a controller and a power supply. A robot can also be provided with

sensors to receive information about the environment [18, 19]. The manipulator (mechanical arm or mechanical unit) is the set of links and joints that position the end effector to perform a given task [19, 20]. An industrial robot is typically supplied with a Teach Pendant which is an input/output unit to allow programming, configuration and monitoring of the robotic system by the user. The controller calculates the robot path and controls the position, acceleration and velocity of every joint of the manipulator in order to perform a programmed task. In general, a robot controller must contain an application program in order to perform a task. An application program is a set of instructions to be carried out by the robot, it is stored in the controller memory [21]. There are three basic ways to generate a robot program: Point-to-Point, Continuous-Path and Off-Line.

The study of the motion of a robot has been divided in two basic parts: Kinematics and Dynamics. Kinematics defines the motion of the robot (position, acceleration, velocity and higher order derivatives) without regard to the forces and torques involved. Using the kinematic analysis, the position and orientation of the end effector can be derived given that the joint variables are known. Also the joint variables can be calculated form a given end effector position and orientation (inverse kinematics) [19, 20]. The spatial rotation and translation concepts are basic for the study of Kinematics and for the current work. An introduction to these concepts is given in Appendix H.

Dynamics, on the other hand, studies the relationship between force and motion [19]. The Dynamic equations describing the forces acting on the motion of the

robot are essential in order to the design and implementation of precise control over the robot actuators.

As mentioned above, a robot can be equipped with sensors which provide it with information about the environment. These sensors can be diverse and their selection depends on the required application. For example, capacitive, ultrasonic or laser sensors can be used to measure distances or detect obstacles. Contact sensors like mechanical switches can indicate the presence of an object. Other sensors can be used to sense pressure, temperature or any other environmental information that can be useful to achieve the desired task in a more accurate way. Cameras are used to give information of the environment in the form of images with the intention of giving the robot a sense of vision, this sensing approach is better known as Machine Vision. In the current work, Machine Vision techniques are used in order to obtain information about the workpiece surface.

2.2.1 On-Line Programming

In the on-line programming method the operator has to “teach” the robot by manually jogging it over the desired path. There are two ways to achieve this: Point-to-Point and Continuous-Path [18, 21].

In the Point-to-Point method, the operator uses the teach pendant to position the robot at several different points on the desired path. The position and configuration of the robot for each point is stored in the robot controller by the

operator using the teach pendant. After all the desired points are stored, the robot can be instructed to move in a continuous way through all the stored points. There are two ways for the robot to move from one point to another. The first one is an independent motion of each axis in order to reach the programmed position, in this way the exact path the robot follows is not known, this type of motion is also known as joint motion. The other way a robot can move from one point to another is through a defined, predictable path such as a line or a segment of curve; this method is known as controlled-path [18].

The Continuous-Path on-line programming method is also known as continuous lead-through or walk-through. In this programming technique, the operator moves the robot trough the desired path while pressing a programming button. While the operator is leading the robot, the controller unit scans and stores the position and orientation of several points on the path. This technique requires the operator to be within the robot's working environment during the programming process [18, 21].

2.2.2 Off-Line Programming

Off-line programming of robots can be understood as a means of generating robot programs without the use of the robot itself [22]. This means that it is not necessary to move the robot through the desired path and "teach" it key points. Using on-line programming usually takes a long time to generate a complete robot program and whilst this may justified if a large volume of identical components are to be produced, it is not a feasible option for one-off welding

situations. Additionally, during on-line programming the robot is out of production for extended periods.

The necessary time and expertise required to put a robot into operation is more evident in some applications than others, which means that in some cases robot automation is too expensive or difficult to implement [20]. Off-line programming can help to reduce the cost of robot automation for certain applications by adding flexibility and reducing the expertise needed to set up a robotic system.

Several different off-line robot programming techniques have been investigated and developed, however some are complex and others are very specific to a single application. It has been claimed that the incorporation of lean, flexible and therefore agile manufacturing philosophies in industry [23] makes necessary the use of equally flexible equipment. This means that robots should be easily and quickly configurable to adapt them to changes in production. This is where off-line programming techniques play an important role since they reduce the programming time of robots compared to the conventional on-line programming methods [23].

Carvalho et al. [22] classify off-line programming techniques according to the level of control the user has on the robot as follows:

- Joint level: The user specifies the motion of each joint of the robot.
- Manipulator level: The user specifies the motion of the manipulator end-effector (Tool).

- Object Level: The user specifies the motions of objects within the robot work-cell.
- Task level: The user instructs the robot to perform a specific task.

Some techniques, like the one proposed, tested and evaluated by Zühlke et al. [4], could be considered to involve more than one of the levels described above, as it is possible to specify joint and manipulator motions and to create macros that involve several single motion instructions, working as a small task.

Many of the task level techniques involve CAD modelling. Although this is a very powerful method for generating programs based on the geometry of a work-piece or the distribution of the robot work-cell, it usually appreciably increases the cost of the system. The determination of the work-piece shape can be done by means of a user interface or by vision-matching. The path can be generated according to predetermined tasks involving predefined work-piece geometry [3] or directly specified by the user in a 3D image which emulates the work cell.

Other techniques allow the user to specify a path on an image acquired by a camera without a previous knowledge of the work-piece, like the one proposed and experimentally evaluated by Ahn and Lee [5]. However, in this particular proposal, the distance of the work-piece is known and the problem applies only to a flat surface. A much more advanced approach is proposed by Nicholson [7] in which the 2D camera image of the work-piece is combined with a Touch

sensing technique (refer to chapter 1.2) adding flexibility in distance and allowing an extension of the technique to inclined planes.

Although these techniques add flexibility to the robot operation, it is still difficult to justify robotic automation in small and medium size enterprises (SME) and where the size of the batches is relatively small due to cost and complexity [2, 24]. When applications are not repetitive, as for on-site repair, it is even more difficult to introduce robots as they must be able to perform different tasks each time at a competitive cost.

2.3 MACHINE VISION

Machine vision may be used to supplement off-line programming as described by Nicholson [7] or to provide a complete off-line programming solution.

2.3.1 Camera Calibration and Lens Distortion

Images provided by a camera are not useful in themselves for obtaining three-dimensional information unless some stereographic technique is used and a camera calibration process has previously been carried out. The process of camera calibration involves the determination of the characteristics of the camera in use (intrinsic parameters) and the relative rotation and translation of the camera to a given reference frame (extrinsic parameters). With this information, it is possible to relate the two dimensional image coordinates obtained to a given world frame in order to calculate three-dimensional data. Although several methods have been proposed for 3D vision systems without a

camera calibration process (for example that proposed and experimentally implemented by Yoshimi and Allen [25]), these usually require additional processing stages and/or place additional constraints on the system. For example, the use of neural networks [26, 27] avoids the necessity of camera calibration but requires a complex and time consuming training stage which in itself is a form of calibration. Guerrero and Sagüés proposed and experimentally tested and un-calibrated camera system for robot navigation [28] based on lines extracted from the images of the environment surrounding the robot. Fofi et al. also proposed an un-calibrated vision system using structured lighting and known reference points; in his technique epipolar geometry is initially used and then a Euclidian reconstruction is performed [29]. Another 3D vision system using un-calibrated cameras is proposed by Gu [30], this employs an iterative method and has constraints in the cameras position relative to each other. Where calibration is used two common methods are:

- The Geometric Method (Tsai method).
- The Planes Method.

Additional concepsts regarding optics that can be useful to the reader may be found in Appendix F.

2.3.1.1 The Geometric Method (Tsai method)

The classic approach to solve the camera calibration problem, proposed by Tsai [31, 32] and widely used elsewhere, is based on solving a group of linear equations that relate pixel coordinates (position in the image) and coordinates

relative to a world frame. The statement and solution of the problem is given below.

The camera is modelled as a simple visual projection problem. A coordinate system with its XY plane parallel to the image plane and the *z*-axis heading outwards from the camera is defined ($x_i y_i z_i$ frame). The centre of this coordinate system is located at the central point of the image on the camera. Parallel to this coordinate frame another coordinate system is defined at the projection focus (O_c). The image plane is the one where the image is formed. In the classical pin-hole camera model this plane is located before the projection focal point; locating this plane after such point does not change the statement of the problem. The projection focus is the intersection point of the $x_i y_i z_i$ frame *z*-axis and a straight line joining a point in the space and its XY image in the camera plane. This situation is illustrated in Figure 2.3.1.1-1.

The image formed at the image plane by an object is a scaled version of the object's XY projection at the projection focal plane, as seen in Figure 2.3.1.1-2; *f* being the distance between the camera plane and the projection focal point.

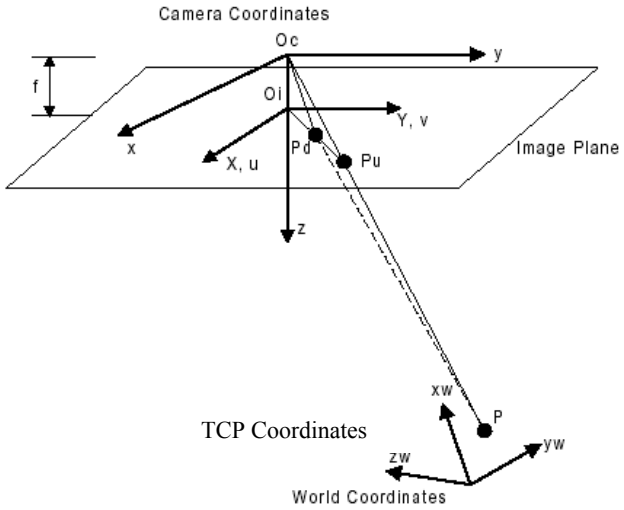


Figure 2.3.1.1-1. Coordinate system definitions. [31]

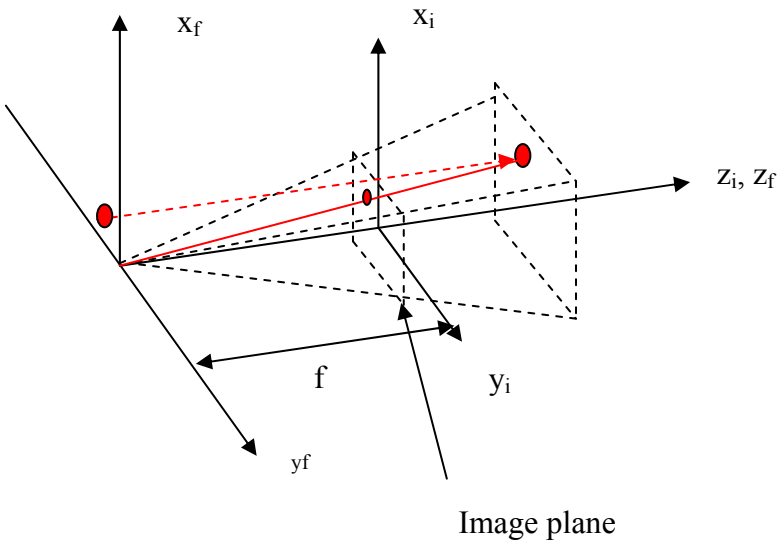


Figure 2.3.1.1-2. Simple camera model.

Considering a point **P** with coordinates (p_{xf}, p_{yf}, p_{zf}) relative to the focal frame as depicted in Figure 2.3.1.1-3, it can be seen that the triangle o_c-p_{zf}-p_{yf} is similar to the triangle o_c-f-p_{yf}. Therefore it can be stated that:

$$\frac{f}{pz_f} = \frac{py_i}{py_f}$$

[Equation 1]

$$py_i = f\frac{py_f}{pz_f}$$

[Equation 2]

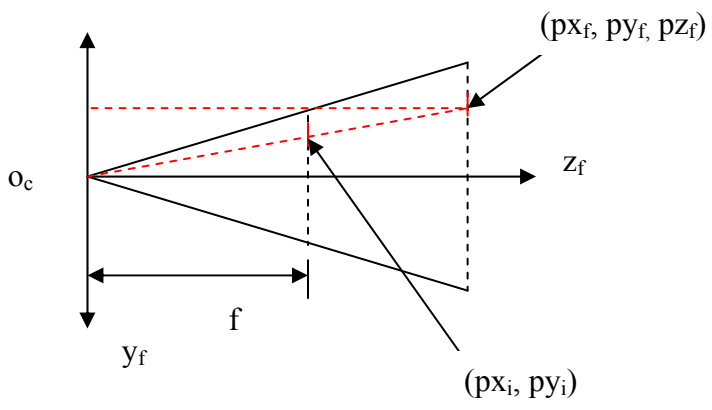


Figure 2.3.1.1-3. Projection of target point in Y-axis of image plane.

Where p_{yi} and p_{xi} are the coordinates of the projection of **P** on the image plane. Similarly it is found that:

$$px_i = f\frac{py_f}{px_f}$$

[Equation 3]

It can also be said that the image coordinates in pixels of **P** (px_c, py_c) are proportional by α to the image coordinates in millimetres (px_i, py_i); therefore the following relationships are obtained:

$$py_c = \alpha_y py_i = \alpha_y f\frac{py_f}{pz_f} = f_y\frac{py_f}{pz_f}$$

[Equation 4]

$$px_c = \alpha_x px_i = \alpha_x f\frac{px_f}{pz_f} = f_x\frac{px_f}{pz_f}$$

[Equation 5]

Where $f_y = f\,\alpha_y$ and $f_x = f\,\alpha_x$. The coordinates of a point **P** in the space, relative to the world frame (px_w, py_w, pz_w) can be related to its coordinates in the focal frame (px_f, py_f, pz_f) by a rotation matrix **R** and a translation vector **T**, as follows:

$$\begin{bmatrix} px_f \\ py_f \\ pz_f \end{bmatrix} = \mathbf{R}\begin{bmatrix} px_w \\ py_w \\ pz_w \end{bmatrix} + \mathbf{T}$$

[Equation 6]

Where:

$$\mathbf{R} = \begin{bmatrix} r1 & r2 & r3 \\ r4 & r5 & r6 \\ r7 & r8 & r9 \end{bmatrix}$$

[Equation 7]

$$\mathbf{T} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

[Equation 8]

The aim is to find the matrices **R** and **T** and the values for f_x and f_y to be able to relate the image and world coordinates. By substituting equations 6, 7 and 8 into equations 4 and 5 the following equations are obtained:

$$px_c = f_x \frac{r1px_w + r2py_w + r3pz_w + t_x}{r7px_w + r8py_w + r9pz_w + t_z}$$

[Equation 9]

$$py_c = f_y \frac{r4px_w + r5py_w + r6pz_w + t_y}{r7px_w + r8py_w + r9pz_w + t_z}$$

[Equation 10]

Dividing equation 9 by equation 10 yields:

$$\frac{px_c}{py_c} = \frac{f_x}{f_y} \frac{r1px_w + r2py_w + r3pz_w + t_x}{r4px_w + r5py_w + r6pz_w + t_y} = \frac{1}{\mu} \frac{r1px_w + r2py_w + r3pz_w + t_x}{r4px_w + r5py_w + r6pz_w + t_y}$$

[Equation 11]

Where $\mu = f_y/f_x$. If the denominators of both sides of Equation 11 are passed to multiply the numerator of the opposite side of the equation, next all the terms

that includes as a factor a component of the rotation matrix are passed to the right side of the equation and then dividing both sides of the equation by t_y the following result is obtained:

$$px_c = \frac{1}{\mu}\frac{r1\,px_w\,py_c + r2\,py_w\,py_c + r3\,pz_w\,py_c + t_x\,py_c}{t_y} - \frac{r4\,px_w\,px_c + r5\,py_w\,px_c + r6\,pz_w\,px_c}{t_y}$$

[Equation 12]

In matrix form:

$$px_c = \begin{bmatrix} px_w\,py_c & py_w\,py_c & pz_w\,py_c & py_c & -px_w\,px_c & -py_w\,px_c & -pz_w\,px_c \end{bmatrix} \begin{bmatrix} \frac{r1}{\mu t_y} \\ \frac{r2}{\mu t_y} \\ \frac{r3}{\mu t_y} \\ \frac{t_x}{\mu t_y} \\ \frac{r4}{t_y} \\ \frac{r5}{t_y} \\ \frac{r6}{t_y} \end{bmatrix}$$

[Equation 13]

As there are 7 variables ($r1/\mu t_y$, $r2/\mu t_y$, $r3/\mu t_y$, $t_x/\mu t_y$, $r4/t_y$, $r5/t_y$ and $r6/t_y$), there must be seven or more equations to solve the linear systems. The equations are obtained by replacing the values of px_w, py_w, pz_w, px_c and py_c, in Equation 13 for N different points in space for which their world coordinates are previously known and their coordinates in the image have been obtained, these points are commonly known as calibration points. N must be much bigger than 7 to obtain

a suitable answer, in fact Sheng et al. [33] demonstrate that camera accuracy improves as N increases.

As **R** is normalised and orthogonal, it s true that:

$$\sqrt{r1^2 + r2^2 + r3^2} = 1$$

[Equation 14]

$$\sqrt{r4^2 + r5^2 + r6^2} = 1$$

[Equation 15]

$$(r1 \quad r2 \quad r3) \otimes (r4 \quad r5 \quad r6) = (r7 \quad r8 \quad r9)$$

[Equation 16]

Where the cross product is represented by $\otimes$. According to Equations 14 and 15 it is clear that:

$$\sqrt{\left(\frac{r1}{\mu t_y}\right)^2 + \left(\frac{r2}{\mu t_y}\right)^2 + \left(\frac{r3}{\mu t_y}\right)^2} = \frac{1}{\mu t_y}$$

[Equation 17]

$$\sqrt{\left(\frac{r4}{t_y}\right)^2 + \left(\frac{r5}{t_y}\right)^2 + \left(\frac{r6}{t_y}\right)^2} = \frac{1}{t_y}$$

[Equation 18]

Using equations 17 and 18 it is possible to find μ and $|t_y|$ from the results obtained by solving the linear system of N equations with the form of Equation 13 and therefore *r1* to *r6*. The third row of **R** is found using Equation 16. Assuming t_y positive and using Equations 9 and 10 it is possible to calculate t_z, f_y and f_x. In matrix form:

$$\begin{bmatrix} px_c \\ py_c \end{bmatrix} = \begin{bmatrix} r1px_w + r2py_w + r3pz_w + t_x & -px_c(r7px_w + r8py_w + r9pz_w) \\ \mu(r4px_w + r5py_w + r6pz_w + t_x) & -py_c(r7px_w + r8py_w + r9pz_w) \end{bmatrix} \begin{bmatrix} \frac{f_x}{t_z} \\ \frac{1}{t_z} \end{bmatrix}$$

[Equation 19]

This linear system is solved for several points using any method for solving linear systems, in a similar way to Equation 13. By doing this the values for f_x/t_z and $1/t_z$ are obtained and consequently the values for t_z and f_x. By using the relationship previously introduced $\mu = f_y/f_x$, the value of f_x can be found. As the image plane and the focal plane are parallel, to determine the correct sign of t_x, it is sufficient to choose a point and calculate its (px_f, py_f) coordinates and compare these with the image coordinates (px_c, py_c). If the point is in the same quadrant in both coordinate systems, then the sign of t_x is correct, otherwise the sign must be changed and the parameters must be calculated again. Equations 11 to 19 are based on the method proposed by Tsai and Motta [31, 32].

Generally, the authors include other parameters in the equations in order to account for lens distortion [32, 34, 35]; the approaches used to address this issue are diverse and the complexity for their calculation differ according to the distortion model considered.

2.3.1.2 The Planes Method

Another interesting approach that is important to consider for solving the camera calibration problem is a geometric approach based on two parallel planes [36-38]. A camera model as shown in Figures 2.3.1.1-1 and 2.3.1.1-2 is assumed; but two parallel planes are used to calibrate the camera as illustrated in Figure 2.3.1.2-1.

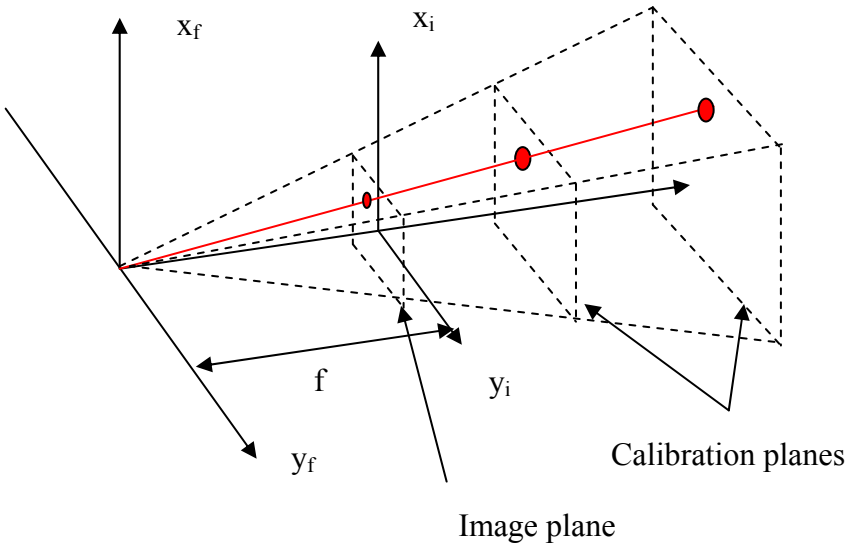


Figure 2.3.1.2-1. Two planes calibration method.

In order to find a relation between each one of the calibration planes and the camera image, several feature points (*N* calibration points) have to be included in each plane and the Equation 20 is used to establish a linear relationship between the image of such points in the image plane and the points themselves. The number of feature points (*N)* must comply with the conditions described in the previous section (2.3.1.1).

$$\mathbf{P}_w = \mathbf{P}_i \mathbf{A}$$

[Equation 20]

Where $\mathbf{P}_w$ is a matrix including the coordinates of the points relative to the world frame for a given calibration plane, i.e. $\mathbf{P}_w = [px_{w1}, py_{w1}, pz_{w1;} \ldots; px_{wN}, py_{wN}, pz_{wN}]$. $\mathbf{P}_i$ is a matrix containing the coordinates of the *N* points in the image, i.e. $\mathbf{P}_i = [px_{c1}, py_{c1}, 1; \ldots; px_{cN}, py_{cN}, 1]$ and **A** is a [3x3] matrix of coefficients to be found. It is clear that the world coordinates of the *N* points contained in $\mathbf{P}_w$ and their projection in the image contained in $\mathbf{P}_i$ must be known in order to obtain the values in the **A** matrix**.** Once **A** is found for each plane, it is possible to find the world coordinates of any point in any of the calibration planes from its coordinates in the image, therefore it is possible to write the following equation for a line in a tridimensional space for each one of *M* given pixels:

$$\mathbf{L}_k = \mathbf{P}_{1k} + \beta_k \frac{\left(\mathbf{P}_{2k} - \mathbf{P}_{1k}\right)}{\left\|\mathbf{P}_{2k} - \mathbf{P}_{1k}\right\|} = \mathbf{P}_{1k} + \beta_k \hat{\mathbf{m}}$$

[Equation 21]

Where $\hat{\mathbf{m}}$ is a direction normalized vector and β_k is a variable; with *k*=1 to *M*. The subscripts 1 or 2 refer to the respective calibration plane. If β_k is replaced by a real number, L_k will take the value of a point in the line described by equation 21. $\mathbf{P}_{2k}$ is a 3x1 matrix containing the *x*, *y* and *z* coordinates of the point in the plane 2 which projects itself as the k^{th} pixel in the camera image, a similar statement can be done for $\mathbf{P}_{1k}$. Now, the projection focus *O*c is the point where all *M* lines intersect; to find it the equation describing the distance (*d*)

from a point (*O*c) to a line in a tridimensional space described by equation 21 must be used [6].

$$d = \sqrt{\left\|\mathbf{T}'-\mathbf{P}_{1k}\right\|^2 - [(\mathbf{T}'-\mathbf{P}_{1k}) \bullet \hat{\mathbf{m}}]^2}$$

[Equation 22]

In equation 22 O_c has been replaced by **T',** the translation vector in world coordinates, as the translation vector (**T**) is no other thing than the 3D coordinates of the projection focus. Since **T'** is the point where all the M lines intersect, **T'** is found just by minimising the square of Equation 22 for all the *M* lines and afterwards solving the linear system. The minimisation process requires the partial derivatives with respect to *x*, *y* and *z to* be determined and equated to zero. It can be shown that the final equation has the form:

$$\mathbf{K} = \mathbf{M}\mathbf{T}'$$

[Equation 23]

Where **K** and **M** are matrices comprising the factors resulting from adding the *M*x3 minimised equations obtained for all the *M* lines. For a more detailed explanation refer to Appendix I. Recalling that **T'** corresponds to the translation vector; once **T'** is found the following equations can be written to obtain the missing parameters [6, 36]:

$$\mathbf{v}_n = \frac{\mathbf{P}_n - \mathbf{T}'}{\left\|\mathbf{P}_n - \mathbf{T}'\right\|}$$

[Equation 24]

$$\mathbf{P}_{cn} = \boldsymbol{\alpha}\mathbf{P}_{in} = \begin{bmatrix} \alpha_x f\mathbf{R}_1 \\ \alpha_y f\mathbf{R}_2 \end{bmatrix}\mathbf{v}_n = \begin{bmatrix} f_x\mathbf{R}_1 \\ f_y\mathbf{R}_2 \end{bmatrix}\mathbf{v}_n$$

[Equation 25]

Where $\mathbf{P}_n$ is any point in the calibration plane for n=1 to N; $\mathbf{R}_1$ = (*r1, r2, r3*) and $\mathbf{R}_2$ = (*r4, r5, r6*) and $\mathbf{v}_n$ is a normalized vector. $\mathbf{P}_{cn}$ is the projection on the camera image of the n-th point $\mathbf{P}_n$ and $\mathbf{P}_{in}$ its respective projection on the image plane. If Equation 25 is applied to N points, then it is possible to find **R**, f_x and f_y just by solving the linear system and using Equations 14 to 16.

Gremban et al. [36] also propose solving the two plane problem including quadratic terms in matrix **A**, it is claimed that this approach produces a better result than using only linear terms. Champleboux et al. [37] claim to have obtained improved results by replacing Equation 20 by a spline interpolation method. Zhang [39] proposes a different approach using planes, where it is required to have two views of the calibration plane with different rotation and translation and assume the calibration plane is located at z = 0.

2.3.1.3 Lens distortion

It is recognised that lenses are a major cause of errors in camera calibration; therefore many techniques have been suggested to address this problem. Some methods just include factors in the set of equations for camera calibration and solve the linear system, others apply iterative calculation methods to estimate the error introduced by the lens such as those proposed by Park and Hong [40]. Distortion is one of the five primary lens aberrations [41] and results in a "*variation in the lateral magnification of points at different distances from the optical axis*" [42]. Distortion can present in two different ways: pincushion

distortion and barrel distortion. Figure 2.3.1.3-1 shows the effect of distortion in a grid-like image.

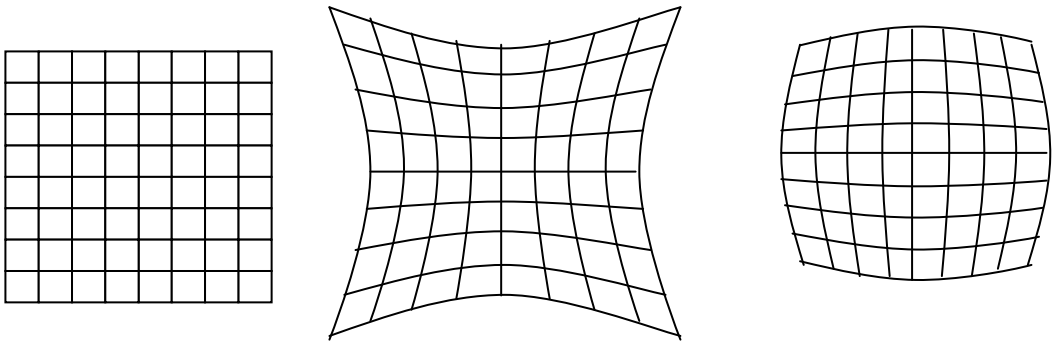


Figure 2.3.1.3-1. Lens distortion.

Although much research has been undertaken relating to modelling lens distortion, Sheng et al. [33] found that when the distortion introduced by the lens is small, including a distortion term in the calibration model can produce worse results than if no distortion was considered. Sheng proposes a method to determine when it is worth introducing lens distortion terms in the camera calibration model. Zhang and Whang [43] also propose a technique to determine if it is necessary to introduce distortion terms and depending on the distortion level and camera resolution a different camera calibration method is used.

2.3.2 Concepts in Digital Image Processing

It is important to review some basic terms related to the characteristic of an image. These are: intensity, brightness, luminance, hue, saturation and chrominance. They are defined by Plataniotis and Venetsanopoulus [44] as follows:

- Intensity: is a measure, in Watts/m^2, of the flow of light power.
- Brightness: is a characteristic associated with a visual sensation of whether an area appears to produce more or less light.
- Luminance: a weighted measure of radiant power.
- Hue: is an attribute of the colour which indicates the dominant wavelength or colour perceived.
- Saturation: is an indication of the purity of a colour. The more white light is present, the less pure the colour is.
- Chrominance: attribute described by hue and saturation.

To represent a colour in a way it can be used in different applications as a source of information different colour models have been created [44]. The following is a review of two colour models, the RGB and the YUV.

2.3.2.1 The RGB (Red, Green and Blue) colour model

This model is one of the best known and most used models. It is used by many hardware devices such as cathode ray tubes (CRT). This model is based in the way the light is perceived by the human eye [44, 45]. The light is received by photo receptors (Cones) present in the eye which are sensitive to Red, Blue or Green Light. The colour information travels to the human brain divided in these three components, the brain then combines the information and produces a full colour image.

Therefore, by combining these three colours, a wide range of colours can be reproduced.

The RGB colour cube is shown in Figure 2.4.2.1-1 for normalized to 1 values. For digital images, the R, G and B values typically range between 0 and 255 (using 8 bit coding).

This colour model is important since it is the one used by many digital cameras including the one utilized in this project.

2.3.2.2 The YUV colour model

This colour model is a luminance chrominance based model [46]. Basically, it consists of two chrominance components and one luminance component. This colour model was initially developed for the transmission of colour television and is used by the PAL (Phase Alternation Line standard) television system. A similar model is used by the NTSC (National Television System Committee) television system, the YIQ colour model.

The Luminance *Y* is given by [47]:

$$Y = 0.299R + 0.587G + 0.114B$$

[Equation 26]

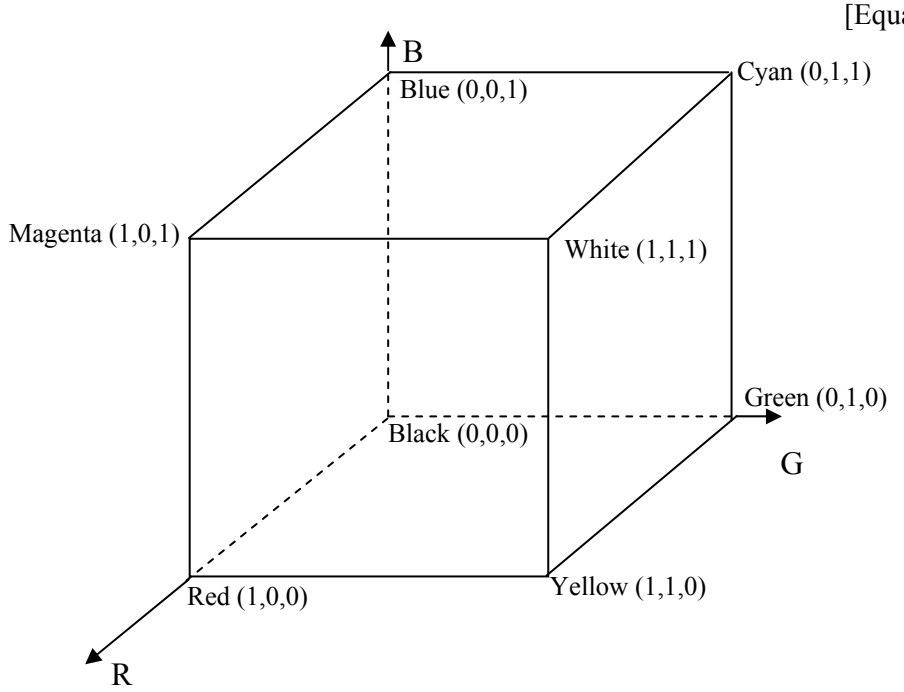


Figure 2.3.2.2-1. RGB colour cube.

The Luminance does not provide colour information about the image; this information is provided by the two chrominance components: *U* and *V*. These signals are defined as:

$$U = \frac{B - Y}{2.03}$$

[Equation 27]

$$V = \frac{R - Y}{1.14}$$

[Equation 28]

The relationship between RGB values and YUV values can be expressed in matrix form as:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

[Equation 29]

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.140 \\ 1 & -0.394 & -0.581 \\ 1 & 2.030 & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

[Equation 30]

In Figure 2.3.2.2-1 the YUV colour cube is shown for Y=0 and Y=1.

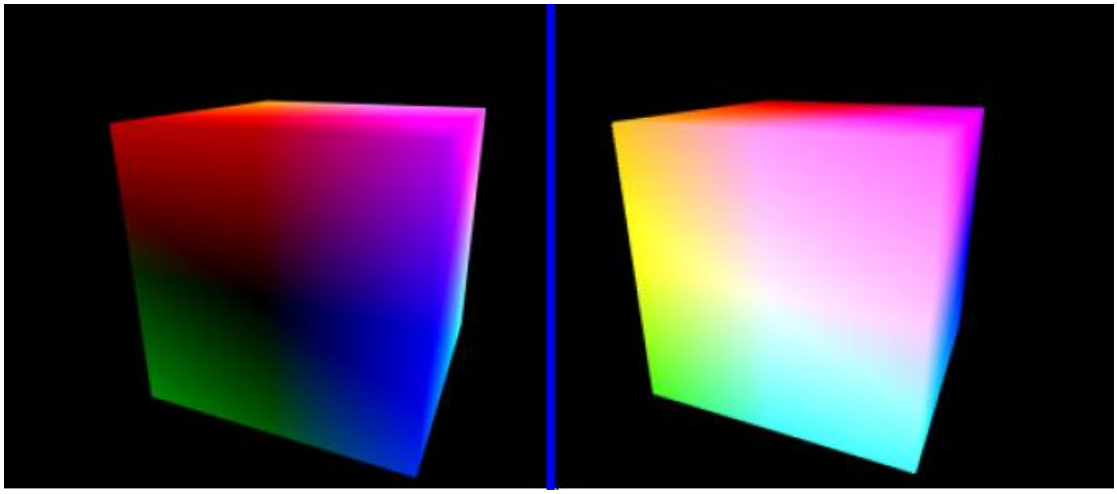


Figure 2.3.2.2-1. YUV colour cube for Y=0 and Y=1. [48]

2.3.3 Structured Lighting

Vision systems in different applications have the purpose of obtaining certain characteristics of an acquired image that will be employed as inputs in order to complete a specific task. In some applications the identification of a characteristic of an object in the image is the main purpose of the use of the

vision system, for example the detection of cracks [12] on a surface or the detection of colours or borders for quality control in production lines [49]. In other applications the information searched for in the image is very difficult to extract and some characteristics of the image are obtained to derive the information required. In this case, the characteristics extracted may be diverse and are dictated by the environment and the processing technique being used. For example, if the information required is the surface profile or the distance of an object it does not matter if lines, borders, colour or other characteristics are extracted as long as they are useful to determine the information of interest.

In the first case above if, for example, the presence of a crack has to be determined then appropriate environmental conditions have to be set in order to be able to extract a crack in the image as no other characteristic is useful. But, for the latter case above, if one characteristic is difficult to extract, then another can be considered. In this case artificial features can be introduced to aid recognition. This is particularly useful in cases where the environmental conditions can change considerably or when the objects has too few characteristics to be recognised [26, 50], then the introduction of artificial characteristics can be useful.

There are several different kinds of artificial features; however structured lighting is a commonly used one. This technique basically consists of the projection of light with a well defined pattern on an object or surface [51]. Although the pattern is known, the way the pattern is reflected by the scene provides information on the scene itself.

The projected patterns can be as simple as a line, a dot, a matrix or concentric circles. For example, Saeed et al. used a dot matrix pattern to scan the three-dimensional surface of a weld pool [52]. Alternatively, these patterns can be more complex such as parallel stripes or sinusoidal patterns of different colours and widths (fringe patterns) [53].

Different light sources are used to project the structured patterns, such as [54]: laser LED's, fibre-optic illuminators, fluorescent light and strobe (Xenon) light. Each of these methods has its advantages and disadvantages and depending on the application one may be more suitable than the others.

It was very common in the past to use video projectors to generate the light patterns, however currently the use of laser light has become more frequent. Solid state laser light emitters are small and therefore suitable for mobile systems; laser light is also monochromatic and less susceptible to environmental changes than the light emitted by a video projector. However, for very complex patterns a laser system can become costly and if different colours need to be introduced, then a video projector may be a better solution. Structured lighting has been used in several applications such as range finders, surface recognition, and weld seam tracking. It can also be used for improving the performance of Neural Networks in machine vision applications according to the work undertaken by Ramachandram and Rajeswari [50].

2.4 TRIDIMENSIONAL SURFACE EXTRACTION

The different techniques used to obtain the tridimensional shape of a surface can be grouped as: non-contact and contact techniques.

2.4.1 Non-Contact Techniques

These techniques do not require the sensing device to establish contact with the surface to be measured. 3D scanner devices can use different sensors in order to extract the coordinates of a surface. These sensors include laser and capacitive devices.

3D scanners using laser technology are based on the triangulation principle. A laser light source projects a laser line on the target surface; the reflected laser light is captured by a receptor which can be a camera or a laser light sensitive device. The relative position of the projector relative to the receptor is known; therefore it is possible to calculate the relative coordinates of the surface using simple geometry and based on the information captured by the receptor of the projected line on the surface, this situation is depicted in Figure 2.4.1-1.

With the 3D scanning technique, as shown in Figure 2.4.1-1, the object must be moved through the laser beam to obtain several contour slices to form the total shape of the surface [55]. The camera used as receptor usually has a high frame acquisition rate in order to analyse several devices as fast as possible. Cameras such as the “Ranger” manufactured by SICK IVP [56] offer a broad

range of options in order to process the image information and are specially designed for the detection and processing of a laser line in an image [55]. The manufacturer also provides the necessary software to process the data. This technology is usually used for inspection in production lines.

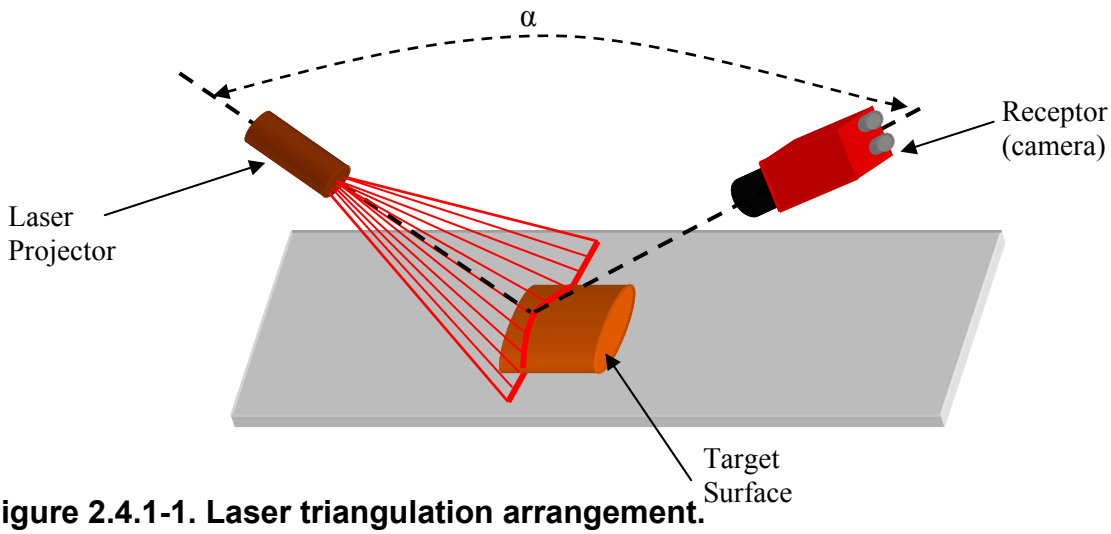


Figure 2.4.1-1. Laser triangulation arrangement.

Laser triangulation is also used in *Coordinate Measuring Machines* (CMM). In contrast to the procedure described above, in these machines the sensor is moved and the surface is stationary. The laser detector and the emitter are usually embedded in a compact enclosure so they can be moved over the target surface. These machines are usually used to produce a digital model of an object in order to manufacture exact copies of it. Renishaw [57] manufactures laser probes for use in CMM machines [58].

Other 3D scanning machines based on laser triangulation rotate the object to be scanned around a fixed axis to generate a 360^{o} model; in this case the sensor

is stationary. Examples of these machines are the “LPX” scanners manufactured by Roland DGA [59].

Other alternative devices are the scanning arms. These are portable CMM devices which can measure the 3D coordinates of a point indicated by the user. They are mostly manual devices composed of a mechanical arm with several degrees of freedom (usually six or seven) and a sensor at the last joint. The measuring arm base is fixed and the sensor is moved by the user to a point on the object to be measured. Several points must be taken in order to generate a model with help of a dedicated software program. The sensor used by these devices can vary but is often a non-contact laser triangulation device. They operate in a very similar way to the inspection technique shown in Figure 2.4.1-1 but instead of moving the target, the sensor is manually moved. FARO Technologies [60] and ROMER [61] are two manufacturers of such measuring arms [62, 63]. According to the manufacturers’ specifications these arms can achieve accuracy of less than 1mm using the laser sensor.

Other laser scanners are designed for mapping the surface of objects located at a greater distance than the above methods. These devices are based on the *time of flight* of the laser or the *phase shift* of a modulated laser beam. Such devices are mostly designed for mapping of an enclosed environment such as a room or a construction tunnel as they can have a horizontal field of view of up to 360^{o} and a large vertical field of view. To achieve this, the sensor rotates around the vertical axis and has a mirror which rotates around the horizontal axis. The generated laser light hits the mirror and then exits the sensor, in this

way it is possible to obtain large field of view. FARO Technologies [60] manufactures *phase shift* devices of this type [63]. The generated laser light is modulated at a high frequency and the difference in phase is measured between the emitted and received light. The phase is proportional to the distance travelled. SICK IVP [56] offers similar sensors but with smaller fields of view, designed for different applications. These sensors use the *time of flight* principle measuring the time that the laser beam takes between the moment it was emitted and the moment it is received back after being reflected by an object [64]. The measurement of time of flight requires a very accurate high speed digital clock able to measure the small fraction of time the light takes to flight from the source to the object and back.

Ultrasonic sensors can also be used for of non-contact 3D measurements. Force Technologies has developed a system based on ultrasonic signals to inspect 3D surfaces [65] This device combines the *time of flight* and *pulse-echo* techniques.

CMM machines can also use capacitive sensors for non-contact measurement instead of a laser detector.

2.4.2 Contact Techniques

Contact techniques require that contact be established between the sensor and the surface. The CMM machines described above can use a contact sensor instead of a laser detector [61, 63]. The sensor has to be moved to a point on

the surface until the sensor establishes physical contact, this process is repeated until a significant number of points are acquired to reverse model the 3D surface of the object. Some of the most common contact sensors include kinematic (mechanical switching), strain, linear variable differential transformers (LVDT) and piezoelectric devices.

For metallic surfaces, a current sensor can be used. A voltage is applied to a metallic end in the sensor, when the sensor touches a metallic surface a current flows and the presence of a surface can be detected. Of course, the object to be measured and the sensor must be electrically connected through a common reference point (ground) for the current to flow. This technique is commonly referred to as *touch-sensing*.

3. PROPOSED SYSTEM DESCRIPTION

The proposed system is composed by different components that together achieve a level of automation that allows the user easily generating welding programs offline through a friendly user interface. Figure 3.1-1 illustrates how the different components link to each other. User input is necessary through the interface which runs in a computer. The specifications of some of the components of the system are given in Appendix E.

3.1 MAIN COMPONENTS

The main components that make up the proposed system shown in Figure 3.1-1 are:

- Digital camera
- Graphic interface and processing software
- Structured light projector
- Welding power supply
- Robotic system

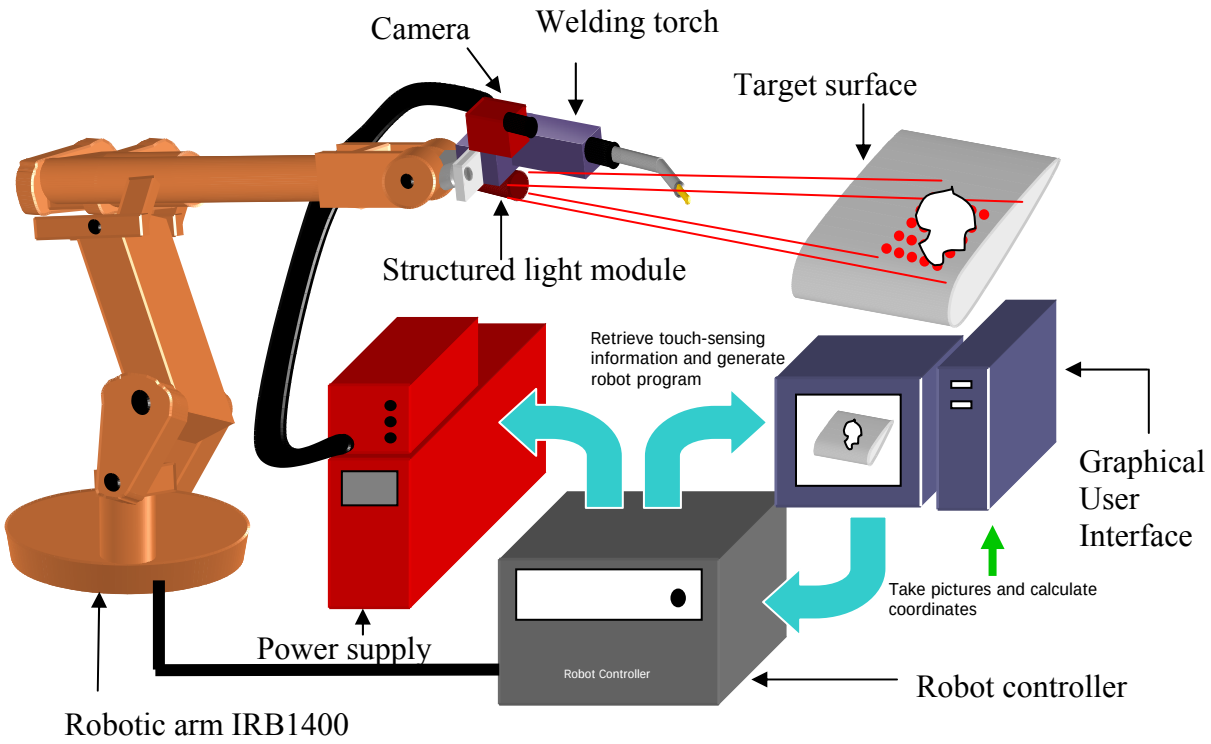


Figure 3.1-1. Diagram of the proposed system.

3.1.1 Camera

The camera used for the acquisition of images was a Marlin 046C manufactured by Allied Vision Technologies, shown in Figure 3.1.1-1. This is a colour CCD camera with a 1/2" imaging device that can acquire up to 53 frames per second and has IEEE1394 connection. It uses the YUV colour mode with 4:2:2 and 4:1:1 compression modes, the RAW format can also be used for monochrome images. The specification of the camera are summarised in Figure 3.1.1-2

Figure 3.1.1-1. Marlin 046C camera used in the system.

The camera is equipped with a 12mm focal length lens able to focus objects from 0.3m and with a minimum f-number of 1.4 (maximum aperture ratio 1:1.4). The camera is attached to the robotic arm and specifically to the welding torch as shown in Figure 3.1.1-3.

The lens is a critical part of the system; a high quality low distortion lens is preferred for better accuracy of the system. The size of the CCD imaging device will determine, in conjunction with the lens, the angle of view of the camera in both directions, horizontal and vertical. It is recommended that a lens of a larger diameter than the CCD format is used in order to avoid defects at the edge of the lens [66].

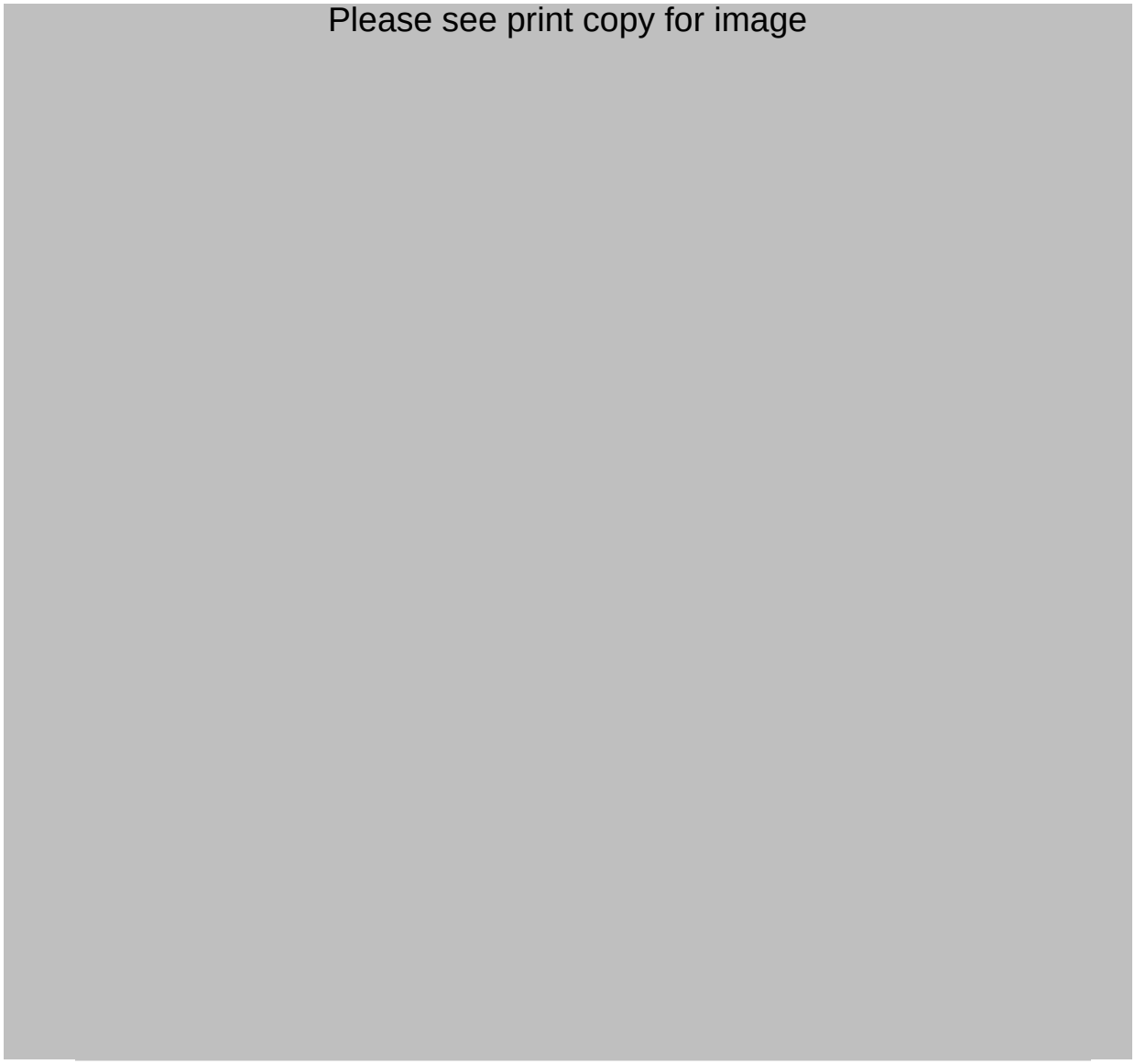


Figure 3.1.1-2. Marlin 046C camera specifications. [67]

Figure 3.1.1-3. Different views of camera attached to the end effector.

3.1.2 Graphic Interface and Processing Software

In order to interact with the user and process the data received from the camera, it was necessary to create a PC software application. The application was developed in Visual Basic 2005 and it incorporates functions such as: sending commands to the camera, viewing the images received by the camera and communications with the robot. This application also served as a data processing and sequence control module of the proposed system.

This interface receives input from the user as well as the visual information received by the camera in conjunction with the input received from the user to generate a complete robot welding program.

The PC used has an Intel Pentium 4 2.8GHz processor with Hyper-Threading technology and 512MB of RAM.

3.1.3 Structured Light

The project was initially developed for welding applications and the surfaces used rarely have significant features which provide the information required to extract 3 dimensional data. To be able to obtain 3D information from such surfaces a laser structured light system was devised to project artificial feature points onto the surface. This laser device provided dot matrix of 7 by 7 points. A more detailed description of this device can be found in Section 4.2.

3.1.4 Welding Power Supply

A Fronius MIG/MAG TransPuls Synergic welding supply was used. This power supply can be controlled through the robot using a robot interface (ROB5000) and the I/O module of the robot controller. This power supply has a touch sensing functionality which enables detection of metallic surfaces. To read the programmed parameters in the power supply, the OPC communications

protocol was used. In this way, the current, voltage and wire feed speed values are updated in the PC application.

Figure 3.1.4-1. Fronius welding power supply.

The CMT (Cold Metal Transfer) welding process is also a feature of this welding system. CMT is a dip-transfer arc technique developed by Fronius and involves the wire motions in the welding process control [68] by incorporating wire transient and current modulation.

3.1.5 Robotic Arm

The robotic manipulator used was an ABB IRB1400; with six degrees of freedom (Figure 3.1.5-2). The communications to the robot controller (Figure 3.1.5-2) are established using Ethernet. To be able to communicate with the

PC, the Factory Ware application supplied by ABB was used. The *Robcomm* communications toolkit provided the tools to enable communication between the robot and the graphical user interface which was developed.

Figure 3.1.5-1. ABB IRB1400 Robotic Arm.

]

Figure 3.1.5-2. ABB SC4 robot controller unit.

4. SYSTEM IMPLEMENTATION

As previously described, a complete system was developed in order to generate a robot program based on visual information for welding repair tasks. In order to do so this, a range of different components had to be integrated and the software and hardware modules had to be created. In this chapter, the main processing stages of the system will be described including the algorithms used and a description of the user interface developed for the proposed system.

4.1 MAIN PROCESSING STAGES DESCRIPTION

The proposed system employs several different processing stages in order to achieve the desired outcome. The first stage consists of the calibration of the camera. The camera needs to be calibrated each time it is removed from the robotic arm and placed back again. This is due to the precision required to achieve accurate results. Once the program generation process has been initiated by the user, it is necessary to project the artificial characteristic points on the target surface and extract the projection of such points from the image provided by the camera. After the 2D coordinates (in pixels) of the characteristic points are obtained, it is necessary to estimate their 3D coordinates and the shape of the target surface.

An area of interest is selected by the user to indicate the system where the welding is going to take place. The user interface provides a drawing tool for

the user to outline the perimeter of the area to be repaired on the captured image using a mouse. The welding path then is calculated, generated and downloaded to the welding robot for the process to start. This process is summarised in the flow diagram shown in Figure 4.1-1.

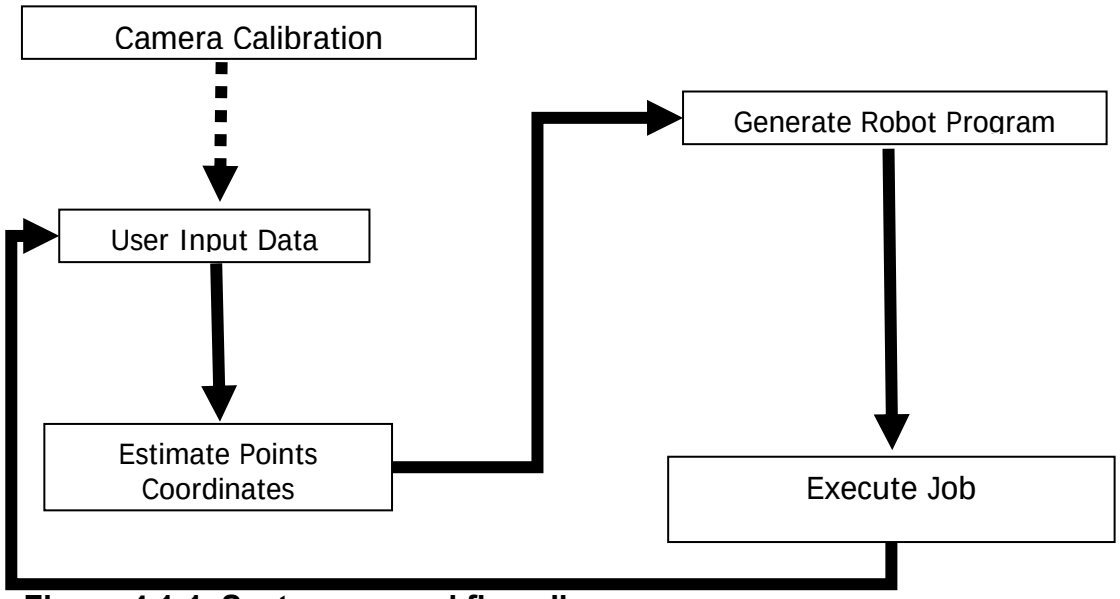


Figure 4.1-1. System general flow diagram.

4.1.1 Camera Calibration

To calibrate the camera an array of 64 black dots in a matrix-like arrangement over a white background was used. The white background provides the contrast required to enable extraction of the dots from the camera image. When the camera calibration sequence is started, the robot is automatically moved to two predefined positions that will provide different sets of dots coordinates, relative to the world frame, which are known. To extract these dots from the image, a clustering technique was used considering only the pixels with RGB characteristics close to complete black; i.e. R=0, G=0, B=0. The idea of the clustering is to group pixels with similar RGB characteristics (black pixels)

that are neighbours to each other. It is done by simply analysing the image pixel by pixel if a pixel is considered to be black then it is verified if any of its neighbours pixels (adjacent pixels above, below, to the right, to the left, and to the four corners) are also black, if so they are grouped as members of the same cluster. After clustering, a set of "objects" or clusters of pixels are obtained, the objects with a size (i.e. number of pixels that forms the cluster) outside a given range are discarded (objects too small or too big). The 3D coordinates of the dots and their 2D coordinates in the camera image are used in the algorithm described below.

The camera calibration techniques described in Section 2.3.1 were tested. However, some modifications to the planes methods were introduced and tested as an alternative calibration method. The first modification was that the relationship between the calibration planes (Equation 20) and the camera was found using a polynomial of third order, but only the linear terms of such a polynomial was used to calculate the calibration of the camera. This means that $\mathbf{P}_i$ in Equation 20 becomes:

$$\mathbf{P}_i = \begin{bmatrix} p_{xc1} & p_{yc1} & p_{xc1}^2 & p_{yc1}^2 & p_{xc1}p_{yc1} & p_{xc1}^3 & p_{yc1}^3 & p_{xc1}^2 p_{yc1} & p_{xc1}p_{yc1}^2 & 1 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} \\ p_{xcN} & p_{ycN} & p_{xcN}^2 & p_{ycN}^2 & p_{xcN}p_{ycN} & p_{xcN}^3 & p_{ycN}^3 & p_{xcN}^2 p_{ycN} & p_{xcN}p_{ycN}^2 & 1 \end{bmatrix}$$

[Equation 31]

The resulting 10 by 3 matrix of coefficients **A** would be as follows:

$$\mathbf{A} = \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ \mathrm{M} & \mathrm{M} & \mathrm{M} \\ j_x & j_y & j_z \end{bmatrix}$$

[Equation 32]

To calculate the *M* points on each calibration plane, only the coefficients *a*, *b* and *j* of each column of **A** are taken into account, therefore Equation 20 becomes:

$$\mathbf{P}_{w1} = \mathbf{P}_i' \mathbf{A}'$$

[Equation 33]

Where:

$$\mathbf{A}' = \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ j_x & j_y & j_z \end{bmatrix}$$

[Equation 34]

$$\mathbf{P}_i' = \begin{bmatrix} p_{xc1} & p_{yc1} & 1 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} \\ p_{xcM} & p_{ycM} & 1 \end{bmatrix}$$

[Equation 35]

$\mathbf{P}_{w1}$ is a matrix containing the *M* calibration points in the first calibration plane.

The modified approach is based on the fact that if in practice there were no errors, the relationship should be totally linear as expected in the theoretical model of the camera. Therefore the aim is, initially, not to find the best average linear model but to try to eliminate the non-linear characteristics of the vision system and model them separately to the linear part.

After the camera parameters are found using the linear terms, the opposite process to calibration is performed to estimate an error function. This means that, using the calibration parameters found, the coordinates of the calibration

points in the image are calculated from the real 3D coordinates using Equations 9 and 10. The error between the calculated image coordinates ($\mathbf{P}_c$') and the extracted image coordinates ($\mathbf{P}_c$) is computed and thereafter a least square method is used to find the coefficients of a third order polynomial describing the error of the extracted points in the image as function of their x_c and y_c coordinates.

$$\mathbf{e} = \mathbf{P}_c - \mathbf{P}_c' = \begin{bmatrix} p_{x1} & p_{y1} \\ p_{x2} & p_{y2} \\ \mathrm{M} & \mathrm{M} \\ p_{xN} & p_{yN} \end{bmatrix} - \begin{bmatrix} p_{x1}' & p_{y1}' \\ p_{x2}' & p_{y2}' \\ \mathrm{M} & \mathrm{M} \\ p_{xN}' & p_{yN}' \end{bmatrix}$$

[Equation 36]

$$\mathbf{e} = \mathbf{P}_i \boldsymbol{\beta}$$

[Equation 37]

The matrix $\boldsymbol{\beta}$ in Equation 37 contains the error coefficients of the camera calibration. When the 3D coordinates of the artificial characteristic points are calculated, these error coefficients may be used to calculate an estimated error which is added to the corresponding extracted 2D coordinates of the characteristic points in the image.

4.1.2 Artificial Characteristic Points Extraction

To extract the laser dots it was necessary to introduce a different algorithm to the one used in the camera calibration stage, since the laser dot characteristics are quite different from those of the black dots. As the laser dots are projected

onto different surfaces, the dot appearance can change depending on the characteristic of the background, in contrast to the constant background used for the black dots. The laser dots, however, have the advantage that they can be projected only when required just by using a digital output of the robot controller. Therefore, it is convenient to take two different shots of the target, one with the dots projected on it and another one without the dots. The pixels where a high change of red value is found are likely to be part of one of the projected laser dots, since the laser wavelength is 653nm which corresponds to visible red. The possibility of change in environmental conditions between the acquisition of the two shots is reduced due to the short time elapsed between them (1 second approximately). Nevertheless, there is some noise present in the images which has to be attenuated; this was accomplished by filtering the images and the use of a weighting value to find the pixels which are more likely to be part of the laser dots in the image. This weighting value includes the difference in the red value between the two images and the proportion of red in the pixel. This value is calculated for each pixel as follows:

$$R_{ij} = (R_{ij1} - R_{ij2}) \times \frac{R_{ij1}}{R_{ij1} + G_{ij1} + B_{ij1}}$$

[Equation 38]

The notation R_{ij} was chosen since the weighting value basically extracts information about the Red component of the pixel only. The subscripts *i*, *j* denote the row and column of the pixel under consideration. *R*, *G* and *B* refers to the red, green and blue values of the pixel and the subscripts 1 and 2 indicates whether the value belongs to a pixel in the laser dots image (1) or the image without the laser dots (2). As indicated by the previous equation, only

the red values of the second shot are used, therefore that image is only filtered in the red plane. A histogram is constructed using these weighting values; this histogram has the form shown in Figure 4.1.2-1.

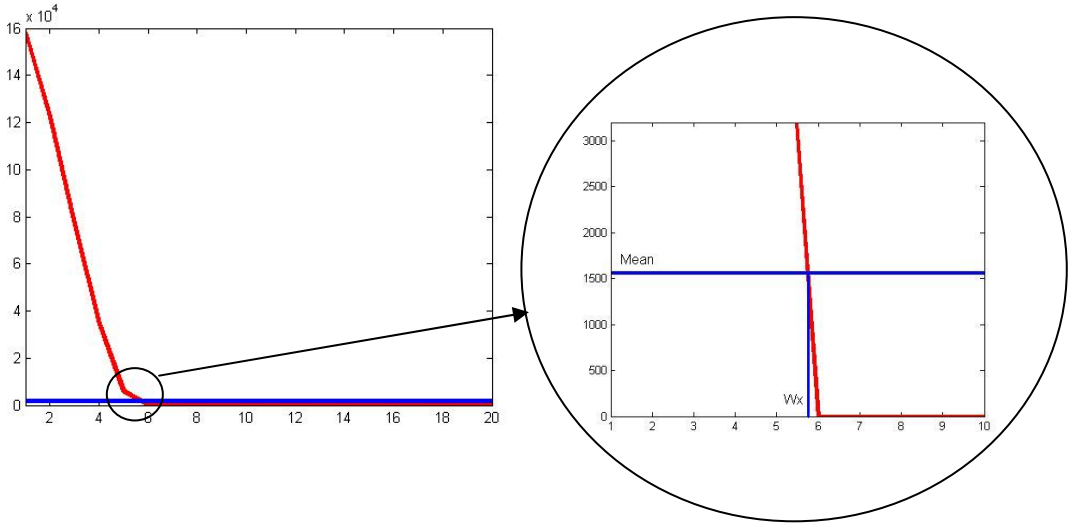


Figure 4.1.2-1. Histogram using weighting values.

The mean value of the histogram is found by dividing the total number of pixels by 256 which is the number of values that R_{ij} can take (i.e $0 \leq R_{ij} < 255$, $R_{ij} \epsilon Z$) and then it is found in the histogram the weighting value W_x (x axis in the histogram) for which the number of pixels is equal to the main value (y axis in the histogram). All the pixels with a value greater than W_x are then grouped as an object if they are adjacent to each other (region growing). Many of these objects are expected to be formed by the laser dots. The objects with less than two pixels are discarded. The maximum red value is found for each object and the objects with the smaller average value between the maximum red and object size are discarded until the desired number of dots is obtained. The location of the pixel with highest red value within an object is assigned as the position of the object and therefore one of the laser dots. This is necessary

because with variation of the angle that the laser light hits the surface and its reflectivity, the shape and size of the extracted dot changes, but the area with higher intensity may be assumed as the real centre of the dot. This extraction process is described in the flowchart in Figure 4.1.2-2.

It was necessary to obtain two sets of points, each one from a different camera position in order to obtain three-dimensional data as will be explained in the next section. Therefore it was required to match the points from the two sets, in other words, to establish which pair of extracted points (one from each set) corresponds to the same physical characteristic feature point. This is known as the “correspondence problem”.

To solve this problem, the fact that the points are arranged in a matrix form was used. Consequently, if the two sets are sorted with a given criteria, the resultant set of points will correspond one to one. The approach used was to sort the points such that the first point is the one located at the bottom right and the last point is the one located at the top left, moving from right to left and then upwards as shown in Figure 4.1.2-3.

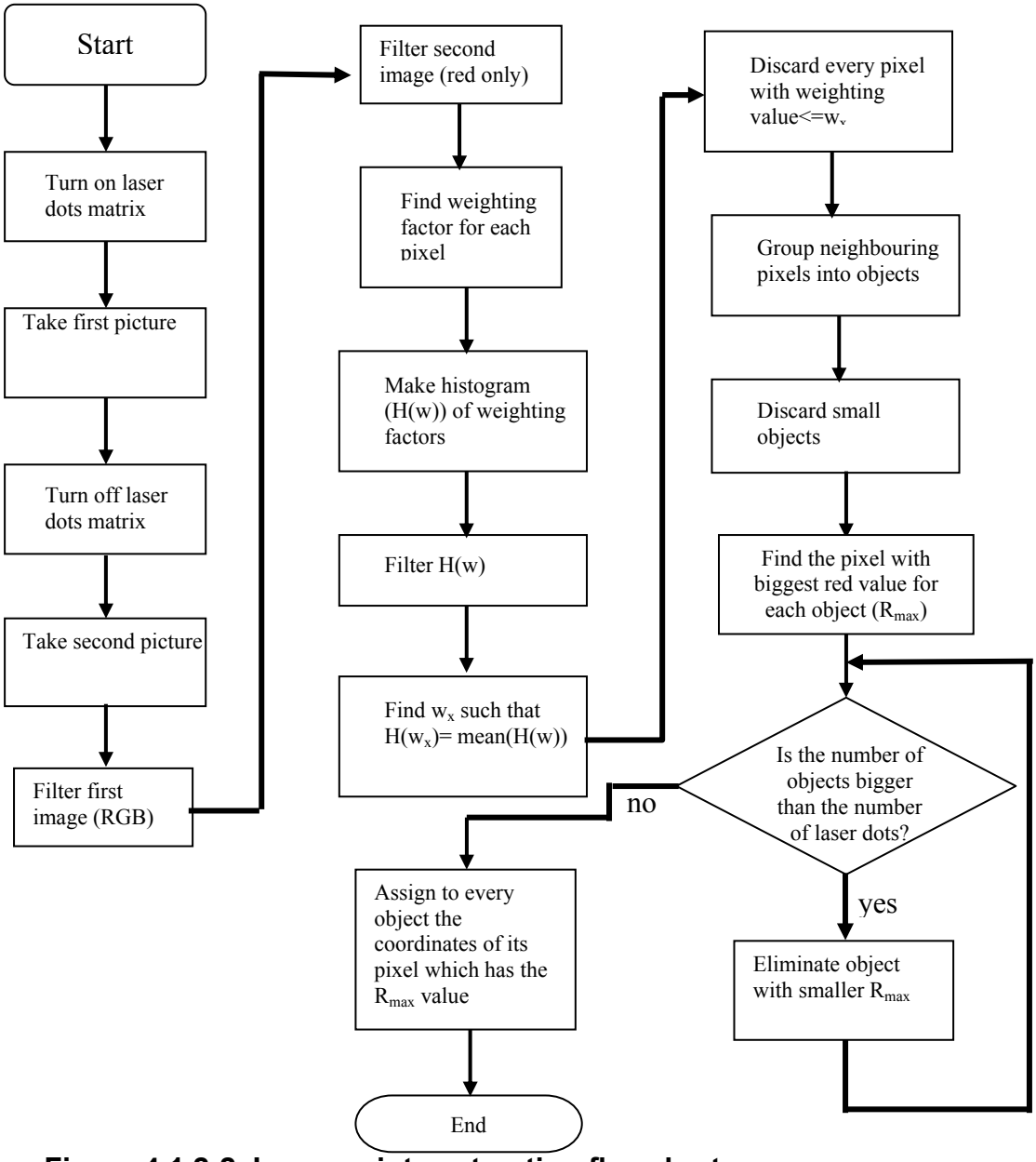


Figure 4.1.2-2. Laser points extraction flowchart.

The problem of sorting the points is trivial if the target surface is flat and parallel to the plane of the camera and the rows and columns of the pattern are aligned with the *x* and *y* axes. However, the projection of the pattern on a curved surface will produce a curved pattern in the image as well (such as that shown in Figure 4.1.2-4) requiring a more complex algorithm. The algorithm used is described in the following section.

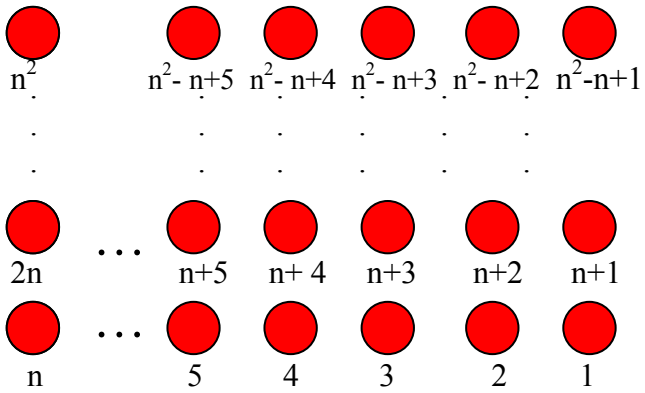


Figure 4.1.2-3. Extracted points sorting.

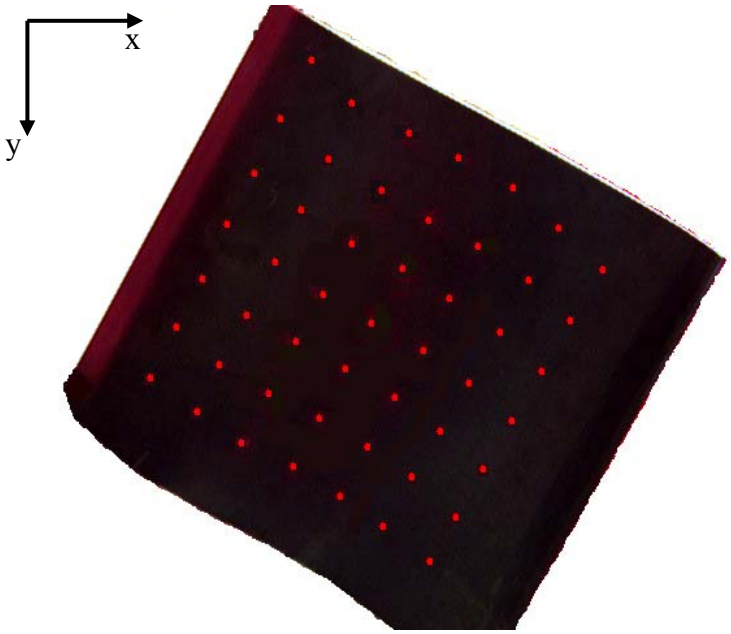


Figure 4.1.2-4. Artificial feature points on a curved surface.

4.1.2.1 Artificial Characteristic Points Extraction

To sort the extracted dots, the first step is to find the dot whose *y* coordinate is the largest. If the pattern is tilted with reference to the camera plane, as in Figure 4.1.2-4, then the dot of the lower corner has been found. If the projection pattern is aligned with the camera axis, the found point could be any belonging to the lower row. However, if the pattern is not aligned, the curvature of the target surface can force the projected points to take a shape such that the middle points of a row have a higher *y* value than any of the two extreme points. To determine if a corner point has been found, its closest neighbours to the right and to the left are found. Then the angle formed by these three points is calculated, if the angle is greater than 2π/3 then it is considered that the found point is not a corner point. In that case, the point to the left is taken and the procedure is repeated until the angle is smaller than 2π/3 or no neighbour to the left is found; then it is considered that a lower corner point has been located.

Once the corner point is found, it is necessary to determine whether it is the left or right corner point. In order to do so, the slope of the line formed by the point and its right neighbour is calculated, the same is done for its left neighbour. If the slope formed with the right point is smaller than the one formed with the left point then it is considered that the point found belongs to the lower left corner and vice versa.

The next step is to find, in order, the other elements of the first row. To achieve this, the neighbour to the left is found if the first point belongs to the right corner, otherwise the neighbour to the right is found. Following, the neighbour of the point previously found is obtained and this procedure continues until all the members of a row are established.

The next step is to find the closest point upwards to the first found point of the current row. After this is done, the previous procedure is repeated until all the points are sorted.

To find the right or left neighbour of a point, the Euclidean distance and the angle information are considered as shown by the following equations:

$$dl = \sqrt{\left(p_{xi} - p_{xj}\right)^2 + \left(p_{yi} - p_{yj}\right)^2}$$

[Equation 39]

$$d\alpha = \frac{\left\|\alpha_{ij} - \alpha_{jj-1}\right\|}{\pi/2}$$

[Equation 40]

$$\alpha_{ij} = \operatorname{atan}\left(\left\|\frac{p_{yi} - p_{yj}}{p_{xi} - p_{xj}}\right\|\right)$$

[Equation 41]

$$\delta_i = \frac{\left(1 - d\alpha\right) + \frac{k - dl}{k}}{2}$$

[Equation 42]

Where p_{xi} and p_{yj} are the coordinates in the *x* axis and y axis respectively of the *i*-th point; similarly p_{xj} and p_{yj} are the coordinates in the *x* axis and y axis

respectively of the *j*-th point point for *i,j* = 1 to *N*, where *N* is the number of dots in the projected laser pattern. The subscript *j* denotes the point the neighbour of which is being found(i.e. the last sorted point); the subscript *i* denotes what point is being evaluated as possible neighbour of the *j*-th point. *dl* is the Euclidean distance between the *j*-th and the *i*-th points. The angle α_{ij} indicates the direction of the trajectory formed by the *i*-th and the *j*-th points; according to the definition of equation 41 the quadrant of the angle is not relevant as if the neighbour to the right of the *j*-th is found then a point is automatically discarded if p_{xi} is bigger than $p_{xj,}$ a similar analysis can be done when finding the neighbour to the left. *dα* indicates the change of direction between the trajectory formed by the point under evaluation (*i*-th) and the last sorted point (*j*-th) and the trajectory formed by the last sorted point and the previous sorted point ((*j*-1)-th). The factor *δ* is a weighting factor which will be decreased if *dl* and/or *dα* increase, the *i*-th point with higher δ_i value will be chosen as the neighbour of the *j*-th point and therefore will be the (*j*+1)-th sorted point. The constant *k* modifies the weight of *dl* in the equation and has to be set to a suitable value which yields consistent results. This procedure is repeated until a full row is completed then the upward neighbour of the first point of the current row is found which will be the first point of the next row. When finding the neighbour of the first point of a row equation 41, i.e. *j* = 1, cannot be used to find α_{jj-1}since the point *j*-1 does not exist. Therefore, the value of α_{jj-1} is found using the last slope calculated in the first step.

When determining the left and right points to locate the corner point, these equations are not applicable either; consequently, the following equations are used:

$$dl = \left\| \left(p_{xi} - p_{xj} \right) \right\|$$

[Equation 43]

$$d\alpha = f\left(\alpha_{ij} \right)$$

[Equation 44]

$$\alpha_{ij} = \left\| \frac{p_{yi} - p_{yj}}{p_{xi} - p_{xj}} \right\|$$

[Equation 45]

$$\delta_i = \frac{d\alpha + \frac{k - dl}{k}}{2}$$

[Equation 46]

$$f(\alpha) = \frac{1 + \sqrt[3]{\left(\frac{1-\alpha}{1+\alpha} \right)}}{2}$$

[Equation 47]

The function *f(α)* penalises slopes greater than approximately 0.85 by giving them a low value. Since *α* can only take positive values, the function output is bound between 0 and 1. As can be seen in Figure 4.1.2.1-1, for an input between 0 and 0.85 the function outputs values between 1 and 0.7; for higher values the output falls sharply until reaching a small value and becomes asymptotic to zero. This function allows more importance to be given to the

change of distance for small slopes, which is useful if the pattern has a small tilt angle relative to the camera frame.

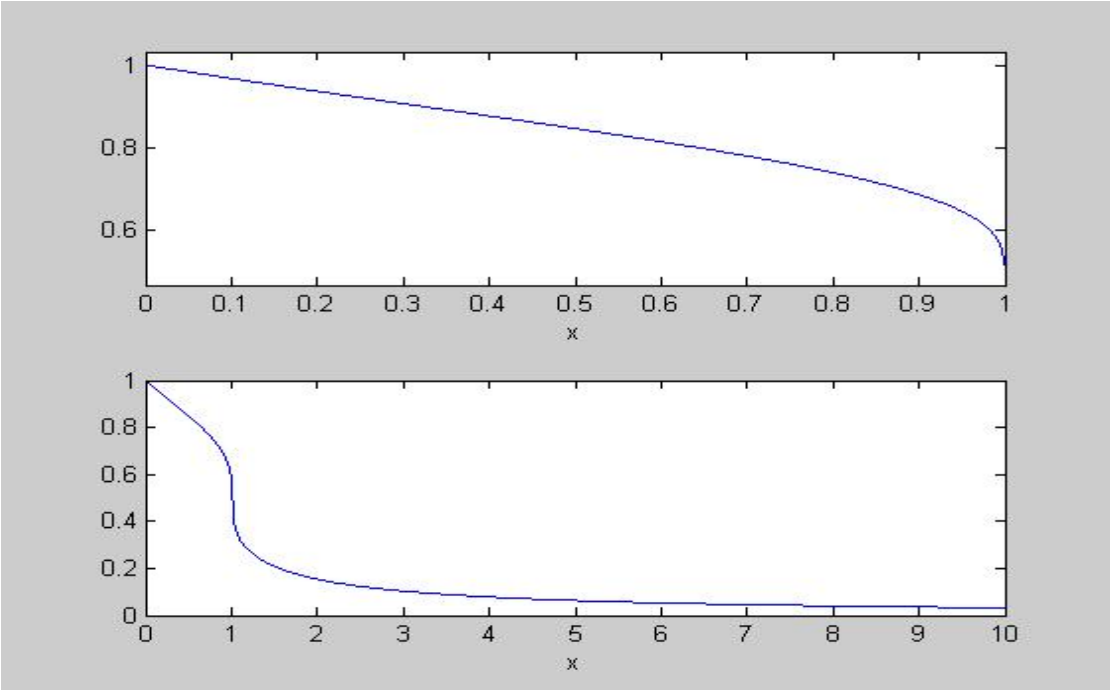


Figure 4.1.2.1-1. Function *f(α)* plots.

To locate the upwards neighbour, the same equations are used except that the slope α_{ij} is calculated as follows:

$$\alpha_{ij} = \left\| \frac{p_{xi} - p_{xj}}{p_{yi} - p_{yj}} \right\|$$

[Equation 48]

The value of *k* used was 640 for finding right or left neighbours and 480 for finding the upwards neighbour. The conceptual flowchart presented in Figure 4.1.2.1-2 summarises the process of sorting the extracted dots from the image.

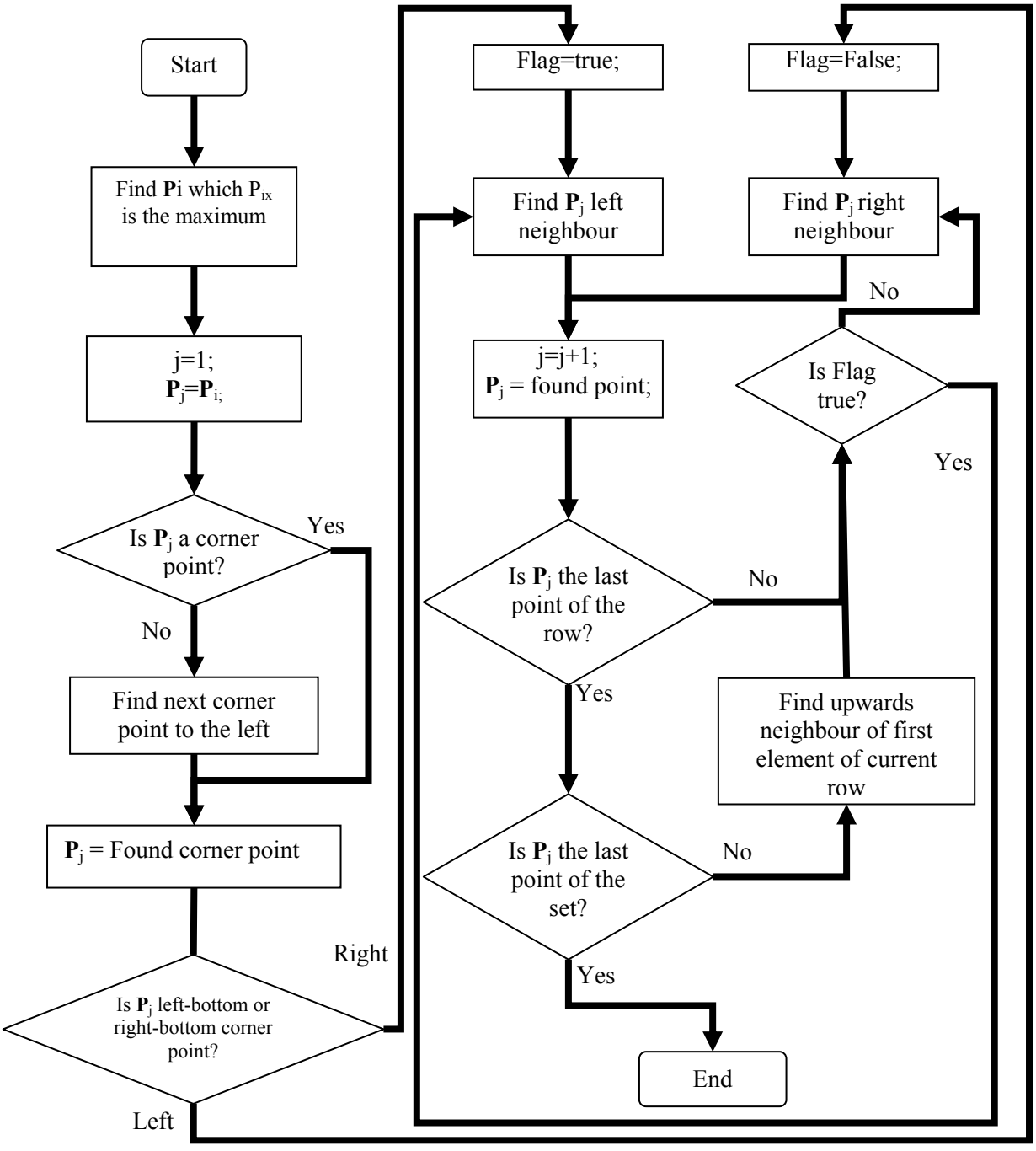
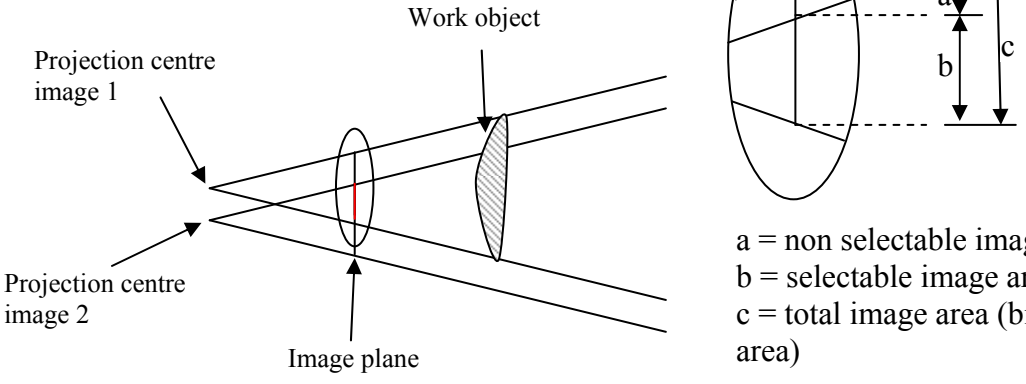


Figure 4.1.2.1-2. Conceptual points sorting process flowchart.

4.1.3 3D Coordinates and Surface Estimation

The algorithm described in the previous section is used to extract the artificial laser characteristic points from two different images and afterwards estimate the 3D coordinates of each point. This is done by using the information given by the change of relative position of each dot from one image to the other.

The fact that at least two images taken from different positions are needed to obtain the 3D data implies that a displacement of the camera has to be made. It is assumed that initially the user jogs the robot to a point where the area of the object to weld is fully visible by the camera. Once in this position, it is possible to take a picture and then moving the camera to another position by applying a predetermined displacement in either *x*, *y* or *x* and *y* direction to take a second picture. These two pictures can be used to estimate the 3D coordinates of the laser dots. This approach would yield a limited selectable area in the image seen by the user on the screen where three dimensional data can be obtained; this situation shown in Figure 4.1.3-1.



a = non selectable image area
b = selectable image area
c = total image area (binocular area)

Figure 4.1.3-1. Binocular projection (without *z* displacement).

In order to avoid limitations to the area which the user can select on the screen, it is possible to move the camera and take the two shots in positions located further back from the original. If these positions are calculated appropriately, there will not be a limitation in the area selectable on the screen as shown in Figure 4.1.3-2. In this Figure it can be seen that the effective binocular area (hatched) formed by the two images is similar to the area seen by the camera in the original position.

To calculate the displacement in *x* (*dx*) and *y* (*dy*) from the initial point the following relationship is used:

$$\frac{dx}{dz} = \frac{screen\ width(pixels)/2}{focal\ lenght(pixels)} \Rightarrow \frac{dx}{dz} = \frac{320}{1530} \Rightarrow dx = dz \times 0.21$$

[Equation 49]

Equation 49 can be verified by the basic geometrical principle of similar triangles. The triangle which sides are *dx*, *dz* and the line segment between projection centre of image1 and the original position (refer to figure 4.1.3-2) is similar to the triangle formed by the original position, the centre of the image plane and middle point of the edge of the image plane.

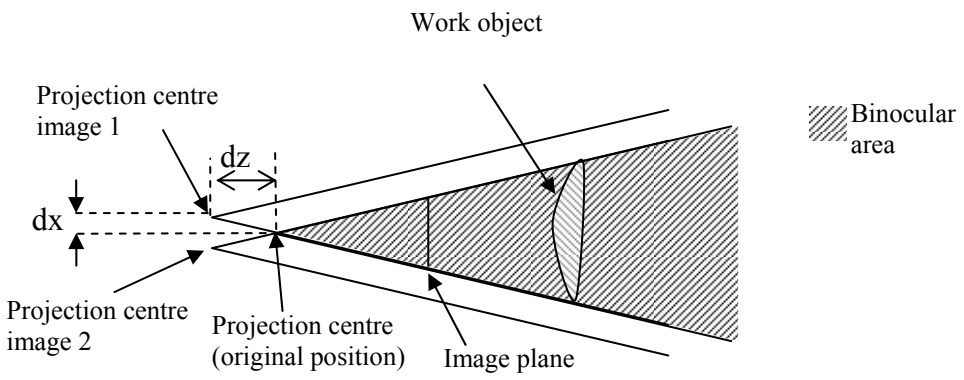


Figure 4.1.3-2. Binocular projection (with *z* displacement).

Similarly, for y direction:

$$\frac{dy}{dz} = \frac{screen\ height(pixels)/2}{focal\ lenght(pixels)} \Rightarrow \frac{dy}{dz} = \frac{240}{1516} \Rightarrow dy = dz \times 0.16$$

[Equation 50]

According to the previous equations, it is clear the change in y is smaller than the change in *x* for a given change in *z* (*dz*). It was found after several tests that a change in pixels of 35mm or more in the position of a test point from one image to another is necessary to obtain an acceptable result when trying to obtain the tridimensional coordinates of such a point. Therefore, it is desirable to have a minimum change of 35 pixels (dy_c) at 1000 mm of distance between the camera and the target (dz_w), which is the maximum defined work distance. Consequently, using the projection equation (Equation 4) the following expression is obtained to find the distance in millimetres in *y* direction from one image to the other (dy_c) :

$$dy_c = f_y \frac{dy_w}{dz_w} \Rightarrow dy_w = \frac{dz_w \times dy_c}{f_y} = \frac{1000mm \times 35}{1516} = 23.08mm$$

[Equation 51]

If we approximate the value of dy_w obtained to be 24 mm and taking into account that dy_w=2*dy*, the following equation can be written to find the necessary displacement in *z* direction (*dz*):

$$dy_w / 2 = dz \times 0.16 \Rightarrow dz = \frac{dy_w}{2 \times 0.16} = \frac{24mm}{2 \times 0.16} = 75mm$$

[Equation 52]

Note that although equation 50 was defined for quantities in pixels it is perfectly valid to use it in millimetres as in Equation 52. By using equation 49:

$$dx = dz \times 0.21 = 75mm \times 0.21 = 15.75mm = 16mm$$

[Equation 53]

Equation 4 can also be used to calculate the 3D coordinates of the artificial projection points. As the distance between the two shots is known, then the distance in *z* can be found as can the *x* and *y* coordinates. This is shown in the following equation:

$$px_{c1} = f_x \frac{px_f + dx}{pz_f + dz} \quad and \quad px_{c2} = f_x \frac{px_f - dx}{pz_f + dz}$$

[Equation 54]

Where subscripts *c1* and *c2* identify the coordinates in camera image 1 and camera image 2 respectively. If pz_f' is defined as:

$$pz_f' = pz_f + dz$$

[Equation 55]

Replacing Equation 55 into Equation 54 the following is obtained:

$$px_f = \frac{px_{c1} \times pz_f'}{f_x} - dx \quad = \frac{px_{c2} \times pz_f'}{f_x} + dx$$

[Equation 56]

$$\frac{pz_f'}{f_x}(px_{c2} - px_{c1}) - 2dx \quad = 0 \Rightarrow pz_f' = \frac{2 f_x dx}{px_{c2} - px_{c1}} = \frac{2 f_x dx}{\Delta px_c}$$

[Equation 57]

Δpx_c is the difference in pixels between the position of one point in the first image and the position of the same point in the second image. Similarly, the same relationship can be found using the displacement of the point in the *y* direction. In theory, the distance *z* established using the information in *x* or in the *y* direction should yield the same result. However, due to the inherent error in the point extraction and the error implicit in the camera calibration, these two values are not equal. Two possible solutions to establish *z* are to average the two values obtained or to choose between them. Experiments were conducted to determine which would give the best result and it was found that, on average, choosing the value found using the information in *x* is the best option. This could be due to the fact that Δpx_c is always bigger that Δpy_c; this would mean that the signal to noise ratiois bigger when using Δpx_c.

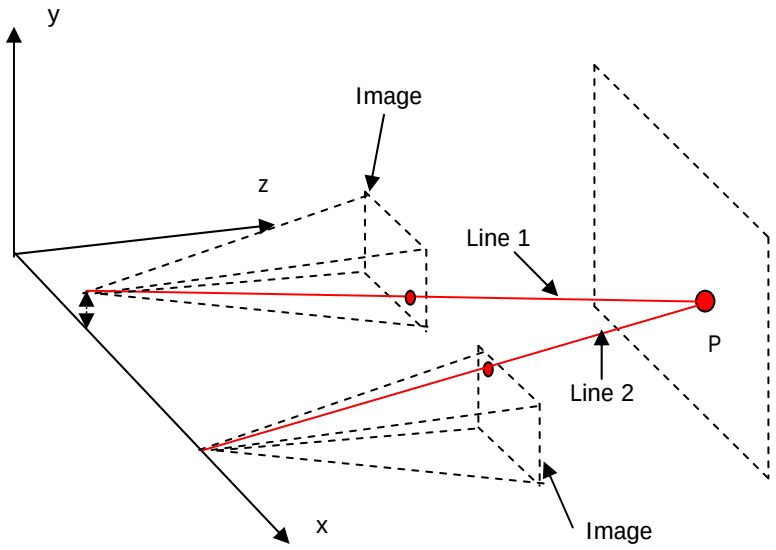


Figure 4.1.3-3. Vectorial approach to calculate the 3D position of a point.

The method described above will be called the *geometrical* calculation method. Another way to find the 3D coordinates of the artificial characteristic points is using a vectorial approach [6]. In order to achieve this, the equation of a line

from the projection focal point passing through the position of the point in question in the image plane for the first image is established. In a similar way another line equation is found for the same point but in the second image, the intersection between the two lines will give the 3D position of the point as shown in Figure 4.1.3-3.

The equation of the line in three dimensions is as follows:

$$l_1 = \alpha_1 \mathbf{v}_1 + \mathbf{F}_1$$

[Equation 58]

$$l_2 = \alpha_2 \mathbf{v}_2 + \mathbf{F}_2$$

[Equation 59]

Where $\mathbf{v}_1$ and $\mathbf{v}_2$ are unit vectors. As already established, these two lines should intersect in space, however, due to the inherent error in the extractions of the points it is possible that they do not intersect. Therefore, to find the point in question it is sufficient to find the values of α_1 and α_2 which minimise the distance between the lines l_1, and l_2. In order to do this, the distance d between a point **P** and a line described by equation 58 must be found using the following expression [6]:

$$d = \sqrt{\left\|\mathbf{P} - \mathbf{F}_1\right\|^2 - [(\mathbf{P} - \mathbf{F}_1) \bullet \mathbf{v}_1]^2}$$

[Equation 60]

Where "•" represents the dot product and **P** is a point lying on line 2 (l_2). Equation 60 is no more than the Pythagoras' theorem where $\left\|\mathbf{P} - \mathbf{F}_1\right\|$ is the

hypotenuse given by the distance between the point **P** and the point $\mathbf{F}_1$ located on the line l_1, one side of the right triangle is given by $(\mathbf{P}-\mathbf{F}_1)\bullet\mathbf{v}_1$ which is no other thing that the projection of the hypotenuse on the line l_1 and the other side of the triangle would be the needed distance *d*. By replacing **P** the following expression is obtained:

$$d^2 = \left\|(\alpha_2\mathbf{v}_2+\mathbf{F}_2)-\mathbf{F}_1\right\|^2 - [((\alpha_2\mathbf{v}_2+\mathbf{F}_2)-\mathbf{F}_1)\bullet\mathbf{v}_1]^2$$

[Equation 61]

To find the value of *α* which minimises the distance, the previous equation is differentiated against α_2 and equated to zero:

$$\frac{\mathrm{d}d^2}{\mathrm{d}\alpha_2} = 2(\alpha_2\mathbf{v}_2+\mathbf{F}_2-\mathbf{F}_1)\bullet\mathbf{v}_2 - 2[(\alpha_2\mathbf{v}_2+\mathbf{F}_2-\mathbf{F}_1)\bullet\mathbf{v}_1](\mathbf{v}_2\bullet\mathbf{v}_1) = 0$$

[Equation 62]

As $\mathbf{v}_2$ is a unit vector, the dot product of this vector with itself is one. Therefore, by solving for α_2 Equation 63 is obtained:

$$\alpha_2 = \frac{(\mathbf{F}_2-\mathbf{F}_1)\bullet\left[\mathbf{v}_1(\mathbf{v}_2\bullet\mathbf{v}_1)-\mathbf{v}_2\right]}{1-(\mathbf{v}_2\bullet\mathbf{v}_1)^2}$$

[Equation 63]

Similarly, it can be found that for α_1:

$$\alpha_1 = \frac{(\mathbf{F}_1-\mathbf{F}_2)\bullet\left[\mathbf{v}_2(\mathbf{v}_2\bullet\mathbf{v}_1)-\mathbf{v}_1\right]}{1-(\mathbf{v}_2\bullet\mathbf{v}_1)^2}$$

[Equation 64]

It can be shown that $\mathbf{v}_1$ and $\mathbf{v}_2$ are given by the following expression [6]:

$$\mathbf{v}_n = \frac{(f_x\mathbf{R}_1 - p_{xn}\mathbf{R}_3) \otimes (f_y\mathbf{R}_2 - p_{yn}\mathbf{R}_3)}{\left\|(f_x\mathbf{R}_1 - p_{xn}\mathbf{R}_3) \otimes (f_y\mathbf{R}_2 - p_{yn}\mathbf{R}_3)\right\|}$$

[Equation 65]

$\mathbf{R}_1$, $\mathbf{R}_2$ and $\mathbf{R}_3$ are the first second and third rows of the camera rotation matrix $\mathbf{R}$ respectively. The subscript *n* is an integer which can take the values of 1 or 2 representing the first or the second point calculation image.

Both approaches were tested in different experiments. Given the results obtained with these methods, it was considered that there was an opportunity to try to improve the results in terms of the error obtained in the 3D coordinates. Therefore, an additional experiment was conducted in which the point's coordinates were calculated using the results obtained by both methods and then performing an average of these; however only the information provided by the *x* coordinate was used from the geometric approach. It was found that this combined technique yielded the best results, therefore this was the method chosen to calculate the 3D coordinates of the points in the present work. The results obtained are analysed in the following chapters.

Once the coordinates of the points are obtained, it is required to include a filtering stage to diminish the effect of "noise" in the data; the effect of noise is illustrated in Figure 4.1.3-4 where the red plot shows how the noise produces a random deviation from the ideal plot (drawn in black); noting that the purpose of Figure 4.1.3-4 is merely illustrative and it does not intend to provide real data. It is supposed that the workpiece has a smooth surface and therefore shouldn't have abrupt changes in its shape.

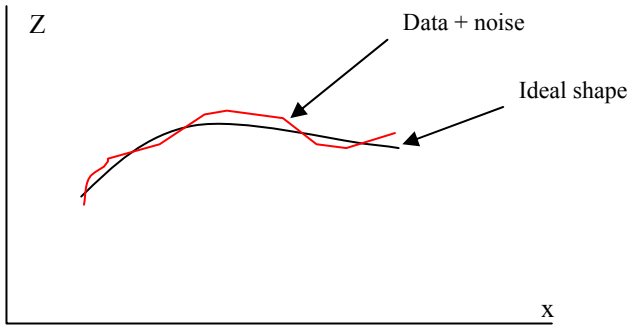
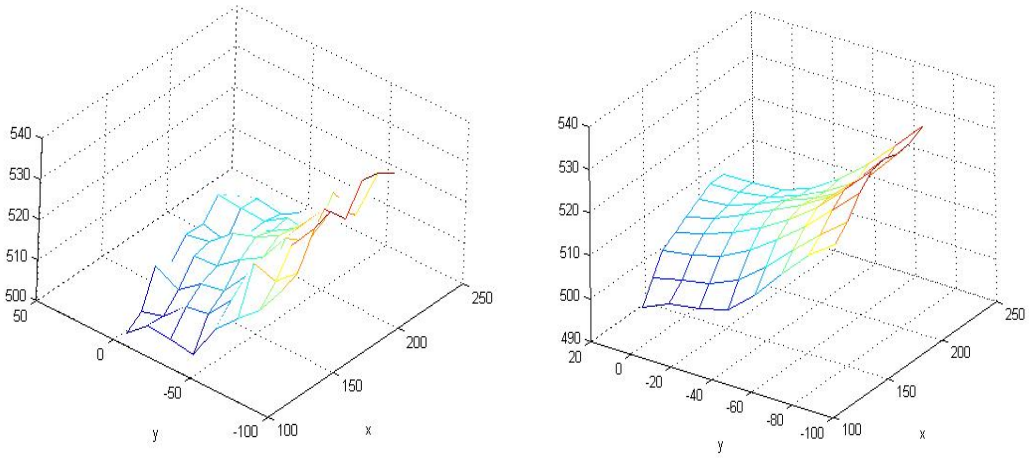


Figure 4.1.3-4. Effect of noise on data.

The largest error obtained is in the *z* axis, with the error in *x* and *y* axes typically being less than one millimetre. This behaviour and the fact that the projected artificial characteristic points are arranged in a grid shape, makes it feasible to two dimensionally filter the data in the *z* axis. A simple low pass digital filter is enough to diminish the effect of noise and random error over the data in *z*. In Figure 4.1.3-5 a 3D plot of extracted data before and after filtering is shown.



The data in all axes is given in millimetres

Figure 4.1.3-5. 3D plot before and after filtering.

The digital low pass filter used was a circular filter in which the bandwidth is constant in all directions. A plot of the 10 point digital filter in the frequency domain is shown in Figure 4.1.3-6, the vertical axis is dimensionless and

represents the gain. A Finite Impulse Response (FIR) filter was used and was calculated using the windowing technique, the Hamming window was employed. An explanation of this method can be found in image processing texts such as Sid-Ahmed [69]. A simple test was conducted to determine the most suitable bandwidth for the current application. Points of data corresponding to a sampling of different smooth surfaces with white noise added were passed through the filter, this was done for different bandwidths of the filter. The bandwidth for which the outputs, on average, were the most similar to the inputs without noise was considered to be the most suitable. By calculating the average of the square of the error of the output signal it was determined how similar was the filter output signal to the input signal without noise; the smaller the error, the more similar.

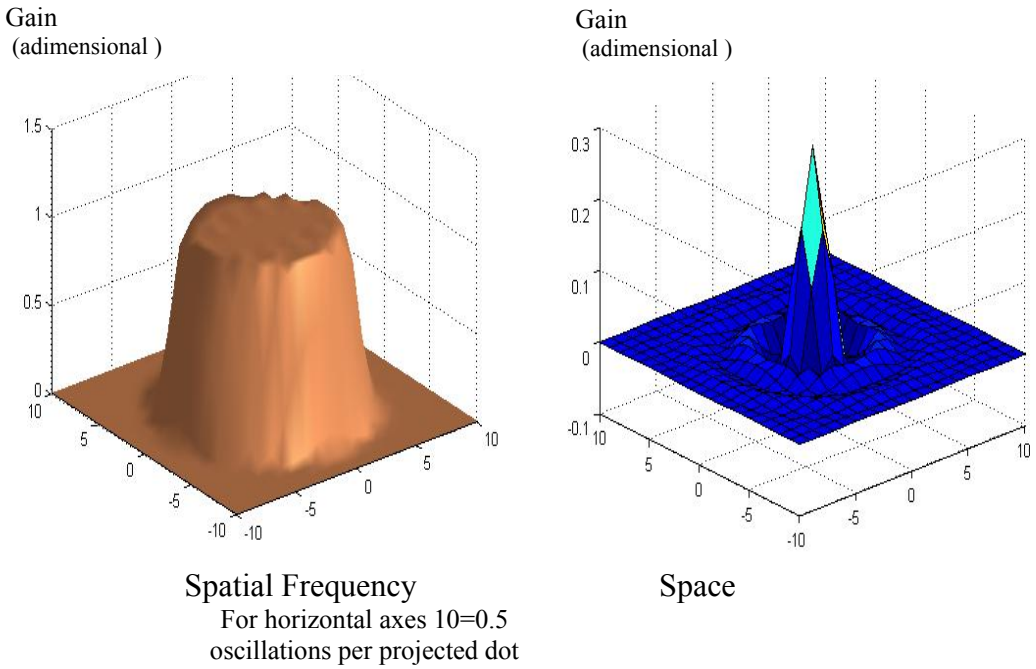


Figure 4.12.3-6. 2D Low pass filter.

After the coordinates obtained are filtered in the *z* axis it is desirable to obtain a mathematical function describing the surface. To do this a polynomial was used and to find it a regression process was employed. Techniques such as splines were not used since they are more suitable when using data points which are considered to be correct (i.e. points with very low error) given that the mathematical function found will pass through these points. However, the points obtained thus far include an error which is not sufficiently small (in *z*) to make them suitable for direct use. On the other hand, the use of regression analysis provides a function that best matches the extracted points. A polynomial regression was used to estimate a function that best describes the surface.

The order of the polynomial in the regression process has to be determined. To do this, the pattern formed by the artificial characteristic points in the image is analysed. As the points are projected in a rectangular grid shape, if the target surface is a plane then the points in the image will form a grid composed of straight columns and rows even if the plane is inclined in relation to the grid projection device. If the surface is curved, then the grid in the image will also be curved. Consequently if the grid is analysed row by row and column by column it is possible to predict the possible polynomial order that best fits the surface in *x* and *y* direction. An analysis of the diagonals of the projected matrix is also performed in order to enhance the results. A pipe resting along the x axis is best described by a plane in the *x* direction and a curve in y direction. When a surface is determined not to be a plane, then a second order polynomial is used to describe the grid (columns, rows or both), if the curves, deviate too much

from the original points in the image, then a third order polynomial is used to perform the regression, otherwise a second order regression is used. To estimate how much the estimated curves deviate from the actual points, the R-squared values for each row, each column and the two diagonals are calculated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{M} (P_i - P_{ri})^2}{\sum_{i=1}^{M} (P_i - \overline{P})^2}$$

[Equation 66]

Where M equals the square root of the total number of projected points, P_{ri} are the estimated points in the image, P_i are the actual points and $\overline{P}$ their average value. By expanding the squared expression and replacing the average value by its definition, this equation can be expressed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{M} (P_i - P_{ri})^2}{\left(\sum_{i=1}^{M} P_i^2\right) - \frac{\left(\sum_{i=1}^{M} P_i\right)^2}{M}}$$

[Equation 67]

The total R-squared value is then calculated for the rows (R_{TR}) and for the columns (R_{TC}) using the average value:

$$R_{TR} = \frac{\sum_{i=1}^{M} R_{iR}^2}{M}$$

Where R^2_{iR} is the R-squared value of the i-th row for 1< i > M and is calculated using only the data in the *x* axis; the R-squared values for the diagonals are also calculated for the values of the points in *x*. If R_{TR} and the R-squared values for the diagonals are equal or greater than 0.999 then it is considered that the chosen polynomial order for the *x* axis is suitable for calculating the 3D coordinates of the artificial characteristic points. A similar analysis is conducted for R_{TC} but the R-squared values of the M columns and the two diagonals are calculated using the data in *y* axis to determine the best polynomial order for the *y* axis.

After the function describing the surface is obtained, (i.e. $z = f(x,y)$) it is desirable to improve the accuracy of this equation. To achieve this, a tactile technique is used to determine the *z* coordinate of selected points. These points are selected from the perimeter of the area selected by the user (refer to next chapter), they include the points with maximum and minimum *z* and two additional complimentary points which allow a good description of the surface with an emphasis on the area to be welded.

The “touch sensing” technique was chosen as it is a standard feature of the power supply being used. Touch sensing is suitable for the detection of metallic surfaces targeted in this research. The technique used applies a nominally low voltage (20V) to the welding wire so that when a metallic surface touches the wire a small current flows. This current indicates the presence of a metallic object and the power supply delivers this information in form of a digital output signal.

After the points to be checked are selected; the robot is moved to the estimated position of one of these points in *x* and *y* direction. For the *z* coordinate a small value is subtracted to account for the possible error included in the coordinate calculation. The value to be subtracted depends on the estimated *z* since the error in this coordinate increases as *z* increases. With the robot head in position the “touch sensing” capability is enabled in the power supply and the robot tool is moved towards the work-piece until it is detected by the “touch sensing” procedure as previously described. As soon as the power supply informs the robot an object has been detected, the robot stops and the current tool position is stored and subsequently read for further processing. This process is repeated for all the other selected points.

After the coordinates of the selected points are found, a new regression is performed. The data provided by these points is added to the data set previously obtained from the artificial characteristic points; however the weighting of the “touch sensed” points is increased by ten times relative to the other data points in the polynomial regression. This is done to emphasize the impact of these points in the solution due to fact that they provide information which is regarded to be true in terms of depth (*z* axis).

4.1.4 Weld Path Generation

In order to generate the welding path, it is required that the user selects an area of interest (i.e. an area containing the damage) on the computer screen. The

objective is to generate parallel weld runs at a constant distance between each other in order to completely reinforce the damaged surface.

The generation of this parallel weld pattern is not straight forward. As the image on the screen is a 2D image, and the work-piece is not necessarily a flat surface parallel to the camera plane, it is necessary to find the real length of the lines forming the perimeter of the area chosen by the user. Figure 4.1.4-1 shows a situation where the distance between the weld runs is constant in the image plane; it can be seen that on a “non-flat” work-piece this distance is not constant.

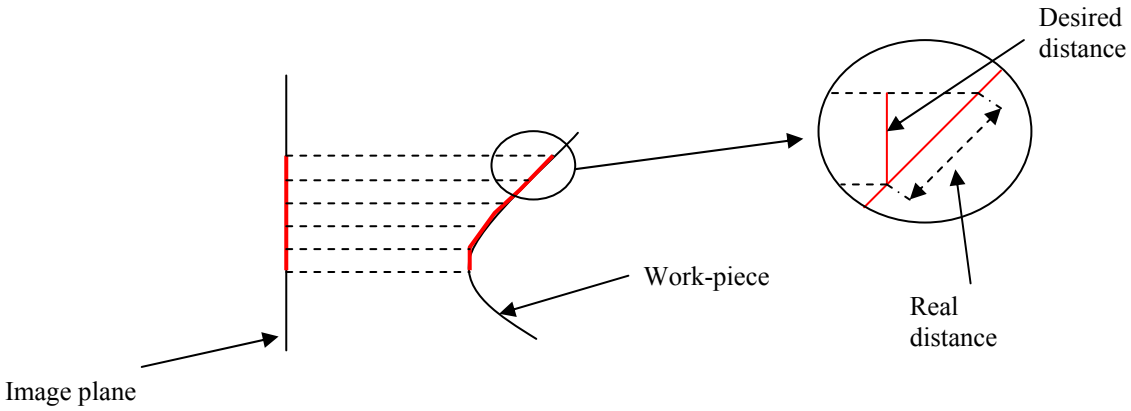


Figure 4.1.4-1. Projection of parallel lines on a non-flat surface.

The first step to generate more accurate weld paths is to establish the correspondence of the points indicated by the user on the computer screen with the target surface. In order to do so, it is necessary to find the intersection of the line formed by the point in the camera plane and the projection focal point with the work-piece as illustrated in Figure 4.1.4-2.

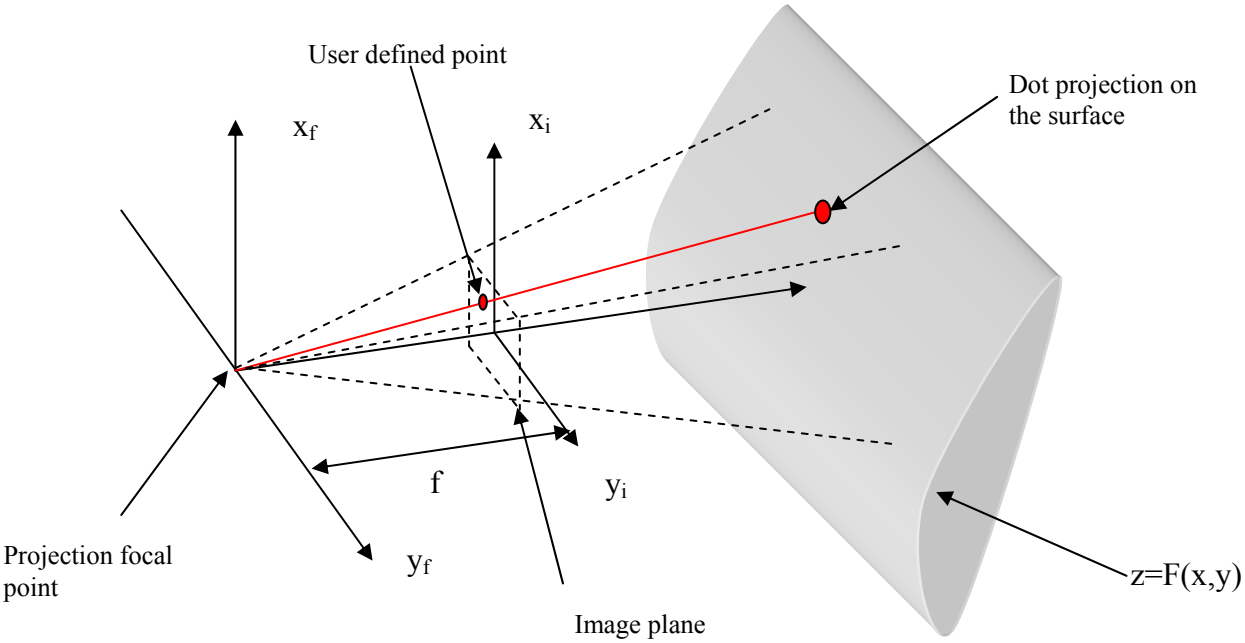


Figure 4.1.4-2. User selected point projected on the work-piece.

Using the basic equations of the projection problem explained earlier in Chapter 2.3.1.1 (Equations 4 and 5) and replacing *px*, *py* and *pz* for *x*, *y* and *z* respectively, the following expression which relate the "real" and "projected coordinates" (i.e. the projection of a point in the work surface on the camera image) can be obtained:

$$x_c = f_x \frac{x_f}{z_f} \; and \; y_c = f_y \frac{y_f}{z_f} \Rightarrow z_f = F(x_f, y_f) = F(k_x z_f, k_y z_f)$$

[Equation 69]

The subscripts c and f indicate camera coordinates and focal frame coordinates respectively as indicated in figure 4.1.4.-2. Where k_x is equal to x_c / f_x, similarly k_y is equal to y_c / f_y. x_c and y_c are the two dimensional coordinates of a point in the camera plane. Therefore, in a third order polynomial form and using Equation 69, z_f would be given by the following expression:

$$z_f = ax_c + by_c + cx_c^2 + dy_c^2 + ex_c y_c + fx_c^3 + gy_c^3 + hx_c^2 y_c + ix_c y_c^2 + j$$

$$z_f = ak_x z_f + bk_y z_f + ck_x^2 z_f^2 + dk_y^2 z_f^2 + ek_x k_y z_f^2 + fk_x^3 z_f^3 + gk_y^3 z_f^3 + hk_x^2 k_y z_f^3 + ik_x k_y^2 z_f^3 + j$$

[Equation 70]

Equation 70 can be rewritten as:

$$0 = j + Az_f + Bz_f^2 + cz_f^3$$

[Equation 71]

Where the coefficients A, B and C are given by:

$$A = -1 + ak_x + bk_y$$

[Equation 72]

$$B = ck_x^2 + dk_y^2 + ek_x k_y$$

[Equation 73]

$$C = fk_x^3 + gk_y^3 + hk_x^2 k_y + ik_x k_y^2$$

[Equation 74]

It is possible to find z_f by solving the previous cubic equation and subsequently x_f and y_f by applying Equation 69. As there are three roots as possible solutions to Equation 70, it is chosen the positive real one. For practical reasons in the remaining of this section z_f, x_f and y_f will be referred as simply z, x and y respectively.

As it was stated above, it is necessary to find the real length (L) of the lines on the objective surface, a good way to do it is to solve the following integral:

$$L = \int_l \sqrt{\left(\frac{dz}{dl}\right)^2 + \left(\frac{dy}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2}\, dl$$

[Equation 75]

For the line $y = mx + b$ in the interval $x_1, y_1 < x, y < x_2, y_2$, being x_1,y_1 and x_2, y_2 the *x* and *y* coordinates of two consecutive points of the perimeter of the area selected by the user. It is relevant to recall that the user defines an area by selecting perimeter points in a 2D plane (the camera image in the computer screen)

If the following relationship for *dz* is defined:

$$dz = \sqrt{dz_x{}^2 + dz_y{}^2} \Rightarrow dz^2 = dz_x{}^2 + dz_y{}^2$$

[Equation 76]

Here dz_x and dz_y are the change of *z* due to *x* and *y* respectively. By substituting Equation 76 into Equation 75 the following expression is obtained:

$$L = \int_l \sqrt{\left(\frac{dz}{dl}\right)^2 + \left(\frac{dy}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2}\, dl = \int_l \sqrt{\frac{\left(dz_x{}^2 + dz_y{}^2\right)}{(dl)^2} + \left(\frac{dy}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2}\, dl$$

[Equation 77]

$$L = \int_l \sqrt{\left(\frac{dz_x}{dl}\right)^2 + \left(\frac{dz_y}{dl}\right)^2 + \left(\frac{dy}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2}\, dl$$

[Equation 78]

From the previous equation, L_x and L_y are defined as:

$$\left(\frac{dL_x}{dl}\right)^2 = \left(\frac{dz_x}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2 \Rightarrow L_x = \int_l \sqrt{\left(\frac{dz_x}{dl}\right)^2 + \left(\frac{dx}{dl}\right)^2}\, dl$$

[Equation 79]

$$\left(\frac{dL_y}{dl}\right)^2 = \left(\frac{dz_y}{dl}\right)^2 + \left(\frac{dy}{dl}\right)^2 \Rightarrow L_y = \int_l \sqrt{\left(\frac{dz_y}{dl}\right)^2 + \left(\frac{dy}{dl}\right)^2}\, dl$$

[Equation 80]

To express the three dimensional length in a two dimensional plane, the total length *L* should be of the form:

$$L = \sqrt{L_x{}^2 + L_y{}^2}$$

[Equation 81]

As *z*(*x*,*y*) has been previously described as a third order polynomial or smaller and as *l* corresponds to the line $y = mx + b$ which corresponds to the equation in focal frame coordinates of the line segment under evaluation that is part of the perimeter defined by the user , then *l* can be replaced by *x*, obtaining: :

$$l = x \Rightarrow dl = dx$$

[Equation 82]

$$y = mx + b \Rightarrow dy = mdx$$

[Equation 83]

$$z = ax + by + cx^2 + dy^2 + exy + fx^3 + gy^3 + hx^2 y + ixy^2 + j$$

[Equation 84]

The components in equation 84 which include *x* are taken into account to find a expression for the portion of *z* that is due to to *x* (z_x) as follows:

$$z_x = ax + cx^2 + exy + fx^3 + hx^2 y + ixy^2 + j_x$$

[Equation 85]

By differentiating z_x with respect to x it is found the change of *z* due to *x*:

$$\frac{dz_x}{dx} = a + 2cx + ey + 3fx^2 + 2hxy + iy^2$$

[Equation 86]

If *y* is replaced by the expression in Equation 83 the following equation is obtained:

$$\frac{dz_x}{dx} = a + 2cx + e(mx + b) + 3fx^2 + 2hx(mx + b) + i(mx + b)^2$$

[Equation 87]

$$\frac{dz_x}{dx} = F_{3x}x^2 + F_{2x}x + F_{1x}$$

[Equation 88]

$$F_{3x} = 3f + 2hm + im^2$$

[Equation 89]

$$F_{2x} = 2c + em + 2imb + 2hb$$

[Equation 90]

$$F_{1x} = a + eb + ib^2$$

[Equation 91]

Similarly for the change of *z* due to y:

$$z_y = by + cx^2 + dy^2 + exy + fx^3 + gy^3 + hx^2y + ixy^2 + j$$

[Equation 92]

By differentiating z_y and replacing *y* by the expression in Equation 83 the following is obtained:

$$\frac{dz_y}{dy} = b + 2dy + ex + 3gy^2 + hx^2 + 2ixy$$

[Equation 93]

$$\frac{dz_y}{dy} = b + 2d(mx + b) + ex + 3g(mx + b)^2 + hx^2 + 2ix(mx + b)$$

[Equation 94]

$$\frac{dz}{dy} = F_{3y}x^2 + F_{2y}x + F_{1y}$$

[Equation 95]

$$F_{3y} = 3gm^2 + h + 2im$$

[Equation 96]

$$F_{2y} = 2dm + e + 6gmb + 2ib$$

[Equation 97]

$$F_{1y} = b + 2db + 3gb^2$$

[Equation 98]

If Equation 83 is substituted into Equation 95 the following equation is obtained:

$$\frac{dz_y}{mdx} = F_{3y}x^2 + F_{2y}x + F_{1y} \Rightarrow \frac{dz_y}{dx} = m\left(F_{3y}x^2 + F_{2y}x + F_{1y}\right)$$

[Equation 99]

Now replacing these in the expressions previously obtained for L_x and L_y:

$$L_x = \int_x \sqrt{\left(\frac{dz_x}{dx}\right)^2 + \left(\frac{dx}{dx}\right)^2}\,dx = \int_x \sqrt{\left(F_{3x}x^2 + F_{2x}x + F_{1x}\right)^2 + 1}\,dx$$

[Equation 100]

$$L_y = \int_x \sqrt{\left(\frac{dz_y}{dx}\right)^2 + \left(\frac{dy}{dx}\right)^2}\,dx = \int_x \sqrt{\left(m\left(F_{3y}x^2 + F_{2y}x + F_{1y}\right)\right)^2 + m^2}\,dx$$

[Equation 101]

Once the values of L_x and L_y are found for every segment of line which makes part of the perimeter of the area defined by the user, a plot of L_y against L_x is made in order to find the set of parallel weld runs.

As previously stated, to make a weld repair in the selected area, parallel weld paths should be performed. As these paths are not necessarily straight but are curved if the surface is curved then it is necessary to specify them in a way the robot can achieve them. The first step in the process of establishing these paths consist of dividing each one of the perimeter length lines obtained in the L_y vs. L_x plot in adequate smaller segments of equal length. The length of each segment will depend upon the welding parameter settings the user has selected on the welding power supply such that parallel lines with an equal distance between each other are generated; i.e. the more welding power used, the wider the weld beads will be and consequently the line segments should be longer. Therefore, it is necessary to find the initial and end points in world coordinates of each line segment. In order to do this, a method based on the technique

used by Nicholson [7] is employed to find the welding lines in the 2-D plane. First the two lines of the perimeter of the area sharing the point with the smaller *y* value in the screen are considered to be the first welding line. The one chosen is that with the smallest slope with respect to the horizontal plane. To obtain the extreme points of the other parallel lines the chosen first line is displaced by the previously established separation and the intersections points of this line with the perimeter lines are found. To find the world coordinates of the intersection points, the following approximation is used:

$$L_x = \sqrt{\Delta z_x{}^2 + \Delta x^2}$$

[Equation 102]

$$\Delta z_x = m_{xz} \Delta x$$

[Equation 103]

Where: Δz_x is the change in *z* due to a change in *x* (Δx).

Since the world coordinates of the extreme points of the perimeter lines have already been found, m_{xz} can be calculated. The *x* coordinate of the extreme points of each line segment are then estimated. To verify that the estimated value of *x* is correct, the length of the line segment is calculated using this value of *x* and compared with the length of the line segment already found. If the match between these lengths is better than 99%, then it is considered an acceptable value, otherwise a new *x* value is estimated just by increasing (or decreasing) the value of *x* by a proportional amount to the error obtained in the length ('rule of three'). The number of necessary iterations is dependent on the curvature of the surface, but as the length is usually relatively small, one or two iterations have been found to be sufficient in most cases. If the slope of the line

is close to infinite, then the previous procedure has to be carried out but for L_y instead of L_x in order to be able to find the coordinates of the points.

This procedure is repeated until the full area has been covered. To generate the final welding path each line is divided in small line segments and the curved line is approximated by small straight lines of approximately 10 mm of length. Nevertheless, if circular motion instructions are used in the robot program, the curved line trajectory can be approximated by smaller curved segments interpolated by the robot controller according to the points indicated in the instruction.

Finally, the tool rotation matrix must be found for every extreme point of each line segment found. This is done to ensure the tool maintains a constant angle relative to the work-piece, i.e. perpendicular to the workpiece, in order to obtain high quality welding. To obtain the rotation matrix, the slope in *x* and *y* direction are found by finding the partial derivatives of the function describing the surface as shown in Equations 104 to 106.

$$m_x = \left.\frac{\partial z(x,y)}{\partial x}\right|_{x=x_i, y=y_i}$$

[Equation 104]

$$m_y = \left.\frac{\partial z(x,y)}{\partial y}\right|_{x=x_i, y=y_i}$$

[Equation 105]

$$z_p = xm_x + ym_y + b$$

[Equation 106]

Being *b* a constant and x_i, y_i the coordinates of the point under consideration. The previous equation (Equation 106) describes a plane tangent to the surface at the point being considered. The notation z_p is used to differentiate between the equation of the surface and the equation for the tangent plane. Therefore it is possible to calculate two vectors on this plane to find a normal vector to this plane as follows:

$$\mathbf{v}_1 = (x_i + k, y_i, z_p(x_i + k, y_i))$$

[Equation 107]

$$\mathbf{v}_2 = (x_i, y_i + k, z_p(x_i, y_i + k))$$

[Equation 108]

Being *k* a constant which can be assigned any value different than 0, in the present work a value of 50 was chosen.

$$\hat{\mathbf{n}} = \frac{\mathbf{v}_1 \otimes \mathbf{v}_2}{\left\| \mathbf{v}_1 \otimes \mathbf{v}_2 \right\|}$$

[Equation 109]

Where $\mathbf{v}_1$ and $\mathbf{v}_2$ are the vectors on the tangent plane and $\hat{\mathbf{n}}$ is the normal vector to the plane. To find the rotation matrix, $\mathbf{v}_1$ or $\mathbf{v}_2$ is taken depending on which is the smallest m_x or m_y respectively. If m_x is smaller than m_y then the rotation matrix is found as follows:

$$\mathbf{R}_1 = \frac{\mathbf{v}_1}{\left\| \mathbf{v}_1 \right\|}$$

[Equation 110]

$$\mathbf{R}_2 = \frac{\mathbf{R}_1 \otimes \hat{\mathbf{n}}}{\left\| \mathbf{R}_1 \otimes \hat{\mathbf{n}} \right\|}$$

[Equation 111]

$$\mathbf{R}_3 = \hat{\mathbf{n}}$$

[Equation 112]

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \mathbf{R}_3 \end{bmatrix}$$

[Equation 113]

Where **R** is the tool rotation matrix.

Whilst these calculations appear somewhat complex and extensive, they are essential steps in establishing accurate weld paths and tool geometry. In practice all these calculations are implemented very rapidly in the software package developed in this work.

4.2 USER INTERFACE DESCRIPTION

A software application was developed to allow the user to provide input to the system and to provide the user with information about the process. Through this software, the user has control over the camera, the welding power supply, and some functionalities of the robot. The program was developed in Microsoft Visual Basic 2005. The full functionality of the system is described below but only a small subset of the software is utilised at the 'operator' level. This operator level interface is described towards the end of this section (Section 4.2.7).

4.2.1 Main Window

The user interface main window is shown in Figure 4.2.1-1. This window has a menu with four sub-menus; they will be described latter in this chapter. To the left, there is a group box named "*Points data*" which contains controls that allow calculation of the 3D coordinates of the characteristic feature points. Another

group of buttons is the "*Calibration*" group that contains two buttons related to the camera calibration process. A third group of controls allows basic camera control actions; defined as "*Camera Control*". The last group of controls is the "*Robot commands*", which allows some basic actions to be sent to the robot.

Two event windows are provided at the bottom centre of the main window, one for camera events and the other for robot events. The camera event window displays any camera errors and open and close camera events. The robot event window displays any error which occurs when a command is issued by the software or there is a communication error between the application and the robot.

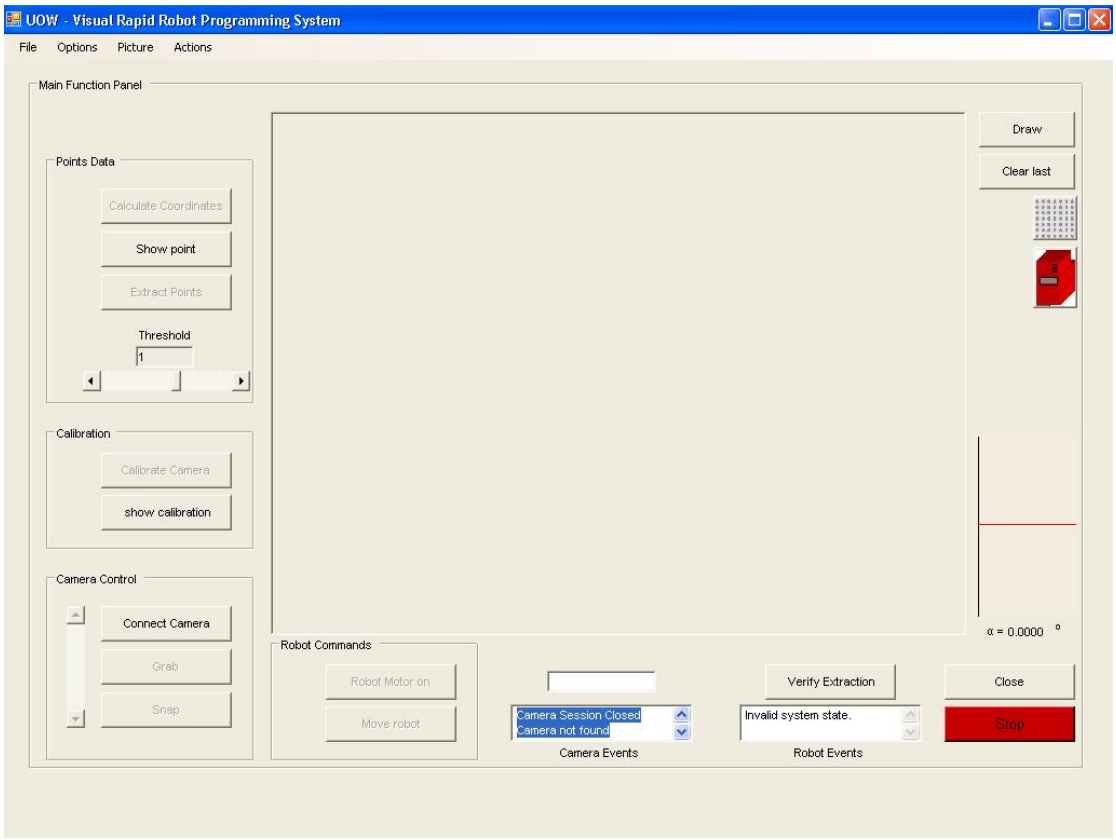


Figure 4.2.1-1. User interface main window.

At the bottom right of the main window, a "*Close*" button is provided, which allows the user to exit the application. A "*Stop*" button is also supplied which allows the user to terminate any action being undertaken by the system and resets the internal variables and logical states to default values.

A graphical tool, which allows control of a rotation of the tool around its *z* axis when welding, is provided at the right edge of the application. This tool looks like a Cartesian coordinate system with a red vector, the orientation of the vector can be modified by the user by "right-clicking" and moving the mouse over the coordinate system.

At the top right, there are two buttons which allow the user to specify the perimeter of the area to be weld repaired. Additionally, a button with a welding power supply drawn on it is available to allow access to a welding parameter configuration window.

There is a large frame at the centre of the window which occupies most of the main window space that displays the images acquired with the camera and is the space in which the user can indicate the perimeter of the area to be welded.

Most of the buttons in the application have a standard size; large enough so that they are easily visible by the user. This feature also gives a good visual appearance to the application.

4.2.1.1 The Main Menu

The top menu in the main window has four sub-menus as shown in Figure 4.2.1.1-1. The first is the "File" menu which has just one item, the "*close*" option. This option allows the user to exit the application and performs the same action as the close button, and the Windows standard "close box" marked with an "X" at the right side of the top bar of the window.

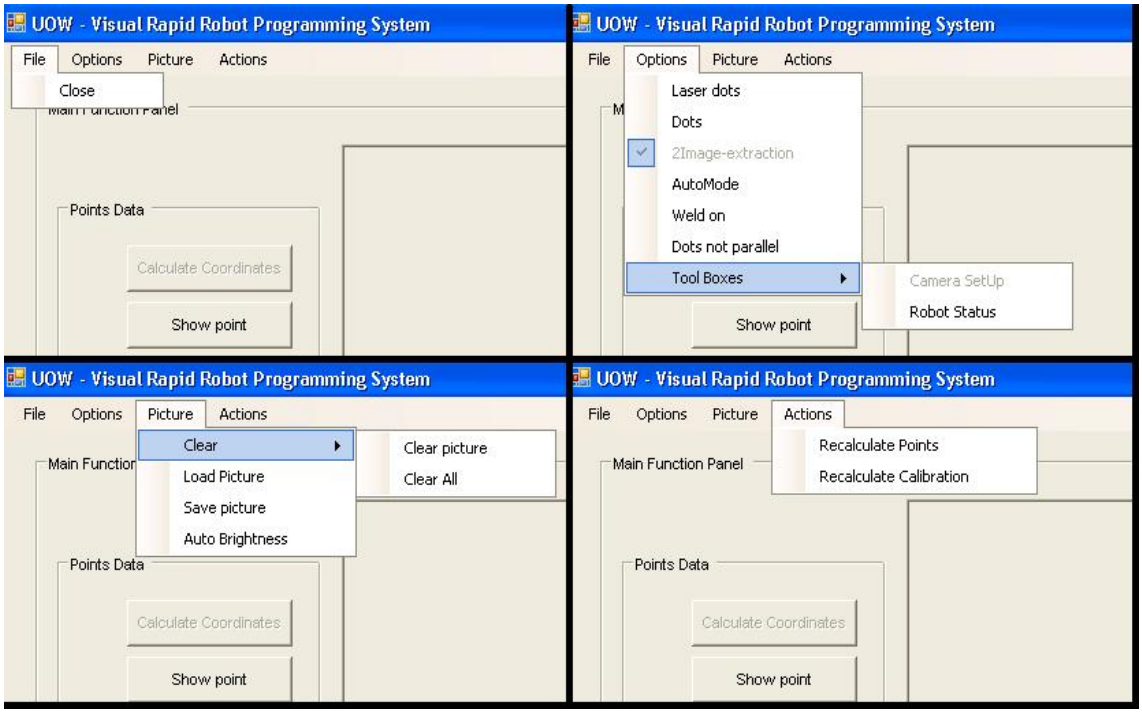


Figure 4.2.1.1-1. User interface main menu.

The second sub-menu is "*Options*". This menu allows the user to enable or disable specific features of the coordinate calculation routine. The first item under this menu is "*Laser dots*", this option indicates to the system if either the red laser dots or black calibration dots are going to be acquired. The second option, "Dots", opens a small modal window which allows the user to specify the number of artificial feature points to be used; this number must have an integer squared root value. The "*2Image-extraction*" option is only enabled when using

the red laser dots and allows the user to choose if the 'two images algorithm' described in the previous chapters should be used or not. The "AutoMode" option enables a full automatic sequence to generate the welding program, if this option is not enabled the user has to instruct the system to perform every step of the process in order to generate the robot welding program.

The next item, "Dots not parallel", will indicate to the system whether the projected pattern has its rows and columns approximately parallel to the camera plane *x* and *y* axes or not. This information is used by the system to choose an appropriate sorting algorithm for the extracted feature points. The last item, "Tool Boxes", gives the user the option to deploy any of the two tool boxes available: "Camera Set Up" and "Robot Status". They will be described below.

The "*Picture*" option is the third sub-menu. This allows the user to perform actions on the image in the main window. The first item under this sub-menu is "*Clear*". This option allows the user to clear the contents of the frame containing the image. Two options are available under this item: "*Clear picture*" and "*Clear All*". The former allows removing the image being shown in the window, the latter clears all images held in memory by the application. It is possible to load a previously acquired image or to save the current image in the window in BMP format using the "*Load Image*" and "*Save image*" options respectively. The "*Auto Brightness*" option will auto-adjust the parameters of the camera to obtain an acceptable intensity level of the image for the calibration stage.

The last sub-menu is "*Actions"*. Under this menu, a recalculation of the camera calibration and the 3D coordinates of the feature points can be carried out by using the "*Recalculate Calibration*" and "*Recalculate Points*" options respectively. These recalculations rely on specific data files the application stores on the hard disk where the points' extraction data from both processes are kept.

4.2.1.2 "Points data" Control Group

This group of controls in the main window is composed of three buttons and a slide bar. The first button is the "*Calculate Coordinates*" button, which launches a window that will allow the user to start the 3D coordinate extraction sequence. The "*Show point*" button opens a window which shows the 3D information of the last extracted points. These two windows will be explained with more detail in following sections.

The "*Extract points*" button will execute the extraction algorithm using an image manually loaded by the user into the application. If the "*2Image-extraction*" option is enabled, the software will request the user to load a second image. With the slide bar below the buttons, the user can modify the threshold value used in the extraction algorithm to manually adjust it for different environmental conditions.

4.2.1.3 “Calibration” Control Group

This control group consists of two buttons related to the camera calibration process: “*Camera Calibration*” and “*Show Calibration*”. The “*Camera Calibration*” button starts the camera calibration process and the “*Show Calibration*” button opens a window which contains the latest camera calibration data.

4.2.1.3 “Camera Control” Control Group

This group allows the user to establish a connection with the camera, start/stop a continuous image acquisition or take a single picture by using the “*Connect Camera*”, “*Grab*” and “*Snap*” buttons respectively. If an error occurs due to any of these actions, an error will be logged into the camera events window at the bottom-centre of the main window. All the communications with the camera are possible thanks to the use of the “NI-IMAQ for IEEE1394” National Instruments library which uses the “IIDC DCAM” standard for IEEE1394 digital cameras.

4.2.1.4 “Robot Commands” Control Group

This group allows command of some of the basic actions of the robot. By pressing the “*Robot Motor on*” button, it is possible to turn on and off the robot servomotors by placing the controller in *Run* mode. The “*Move Robot*” allows the user to move the robot to a specified *x*, *y* and *z* coordinate position. When this button is pressed, a window similar to the one shown in Figure 4.2.1.4-1 is

launched. This window has three text boxes labelled “x”, “y” and “z” where the user can specify the desired new location of the tool. The user can choose between “relative” and “absolute” displacement by selecting one of the three radio buttons located to the left side of the window. If “absolute” displacement is selected, the robot will move relative to a predefined work-object used for calibration. If the displacement chosen is “relative”, the robot will move to the specified coordinates relative to the current position. The orientation of the torch can also be specified by selecting one of the two radio buttons in the “orientation” control group. If “tilted” is chosen, the final rotation of the tool is a fixed tilted rotation used in the camera calibration process. If “parallel” rotation is selected, the tool is aligned with the work object coordinate frame. If the “Reference Position” radio button is selected, the position data supplied by the user will be ignored and the robot will move to the reference position (first position) of the last extraction process run. If no extraction process has been run since the last time the application was launched, this option will be disabled as this reference position is stored in volatile memory.

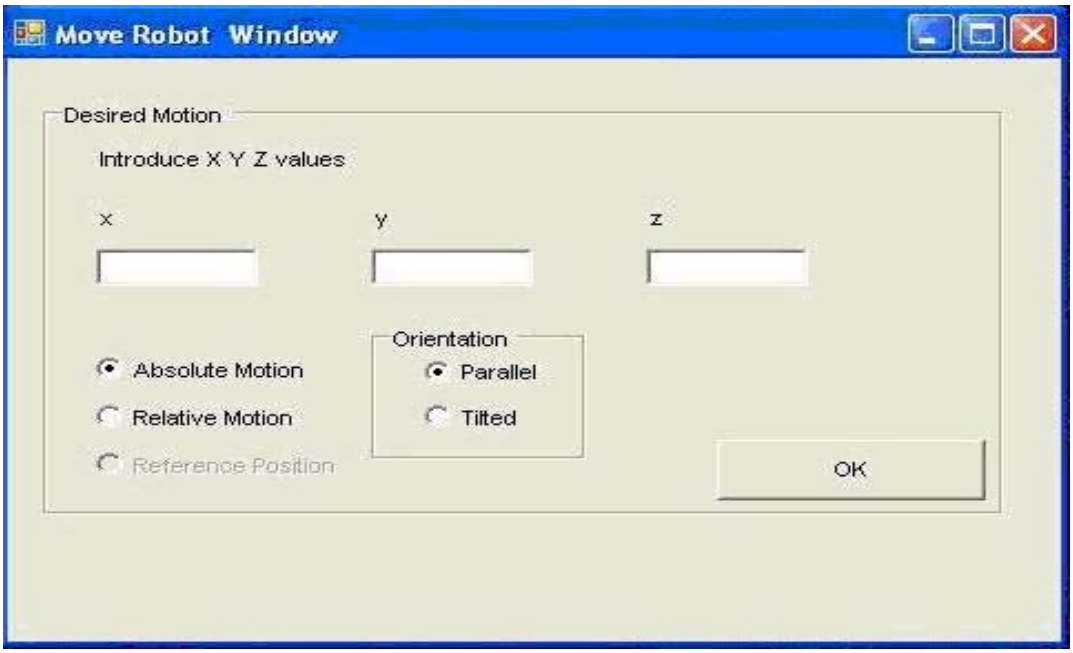


Figure 4.2.1.4-1. User interface "Move Robot" window.

4.2.2 Welding Parameters Window

The "*Welding Parameter Window*" allows the user to configure the welding values in the welding power source. The communications with the Fronius power supply is carried out in two different ways, the first is indirectly through the I/O module of the robot controller and the second is a direct serial connection which uses the OPC protocol at a higher level. Figure 4.2.2-1 shows the interface screen.

The main menu of the window, shown in Figure 4.2.2-2, has one sub-menu, "*Actions*" with two options: "*Download Program*" and "*Close*". The "*Download Program*" allows the user to download a robot program file with ".prg" extension to the robot controller floppy disk. When the user clicks on this item, a file-open dialog window opens for the user to select the file to download; after the file is selected, a modal dialog input window opens to request a destination file name (on the robot controller floppy disk), no extension should be specified. The "*Close*" option closes the window.

In this screen there are six text boxes at the top left where the wire feed speed, current, voltage, filler metal, wire diameter and shielding gas information is shown. The first three parameters correspond to the actual programmed values in the power supply. These parameters are obtained via OPC connectivity with the power supply. To establish this connection, the Fronius "LocalnetOPC"

server is started by the application when it is loading. When the welding parameter window is opening, a connection is established with the OPC server so that every time any of these three parameters changes (current, wire feed speed or voltage), its value is updated on the screen.

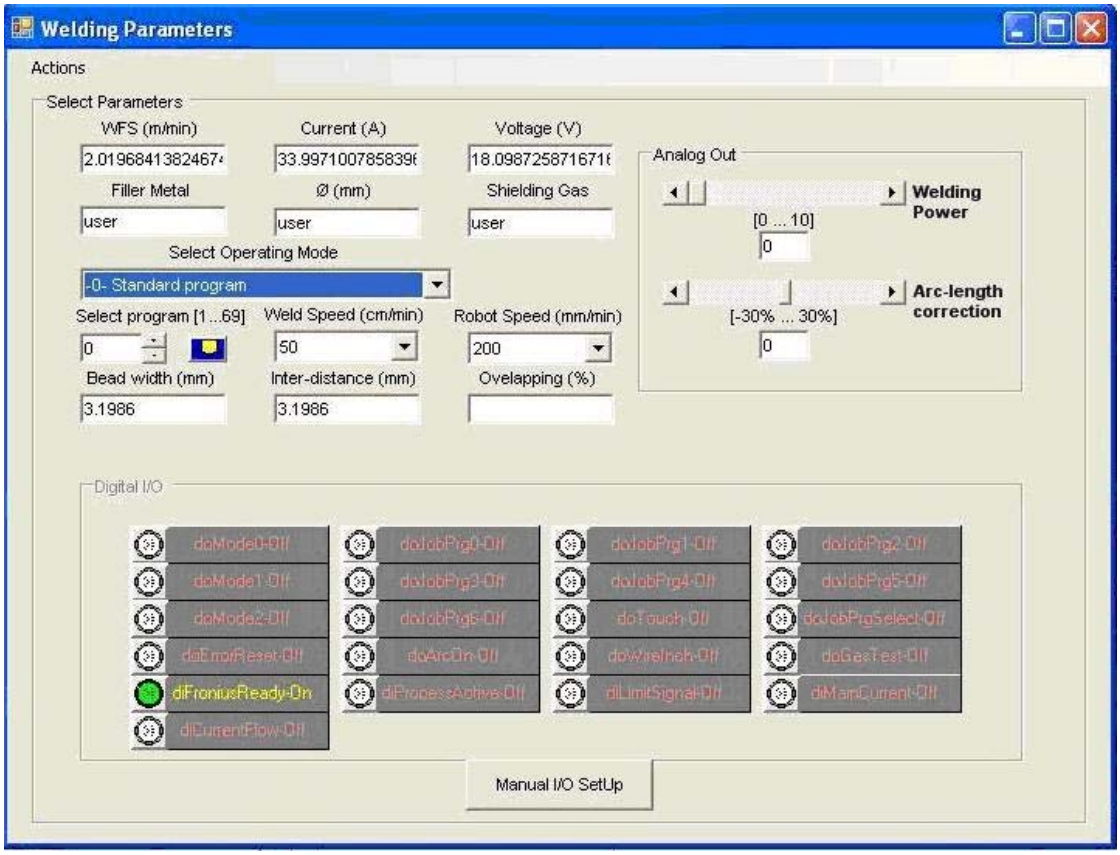


Figure 4.2.2-1. User interface welding parameters window.

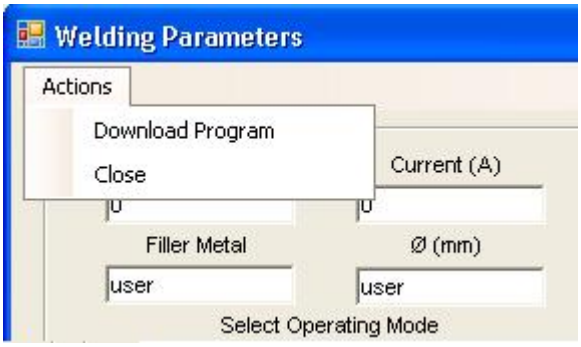


Figure 4.2.2-2. User interface welding parameters window menu.

The next three parameters (filler metal, wire diameter and shielding gas) are associated with a specific program. When a welding program is selected in the power supply using the user interface, the application loads these parameters from an excel file containing the data related to all welding programs in the power supply.

Just below these text boxes, a drop down menu is provided for the user to choose the desired operating mode of the welding power supply. Eight different operating modes are possible: Standard Program, Pulsed Arc Program, Job Mode, Parameter Selection, Manual, CC/CV, TIG and CMT. On selecting any of the program modes or job mode, the user can specify the program/job number using the text box located just below the operational modes drop down menu. The user can also introduce the desired welding speed and robot speed using the relevant drop down menus. The robot speed is the speed the robot will move if the program is executed step by step. The overlapping percentage of the weld beads can also entered to the system by entering the value in the relevant text box, the software estimates the weld bead width and the inter-distance between beads and presents them to the user through the dedicated text boxes in this window.

To the right of the screen, two scroll bars allow control of the two analogue outputs of the robot controller I/O unit. The parameters that can be modified through these controls depend on the operational mode chosen. For most of the operational modes, these parameters are weld power and arc-length correction, for manual mode they are wire feed speed and welding voltage, for

CC/CV mode they are wire feed speed and welding current, for TIG mode they are arc-length correction and welding current and for manual mode the controls are disabled. These different uses for the same analogue output are characteristics of the Fronius hardware, and the software only modifies the labels on the screen in order to give an indication to the user about what parameter is being modified.

In the lower part of the screen, there is a group of control buttons called "*Digital I/O*". These controls each have an associated pilot light. Each button allows toggling the corresponding output and the pilot light indicates the current status of the input/output. To create these controls, the "*ABB RobComm ActiveX Library"* was used. The "*Manual I/O Setup"* button is provided at the lower-central section of the screen, by selecting this button the user can click on one of the I/O buttons. This feature was included to avoid the user accidentally changing the status of one digital output, thus producing an undesirable action.

To modify the welding parameters in the welding power supply, the digital and analogue outputs of the robot controller are used. Additionally a unit provided by Fronius plays the role of interfacing these signals with the welding power supply. This situation is depicted in Figure 4.2.2-3.

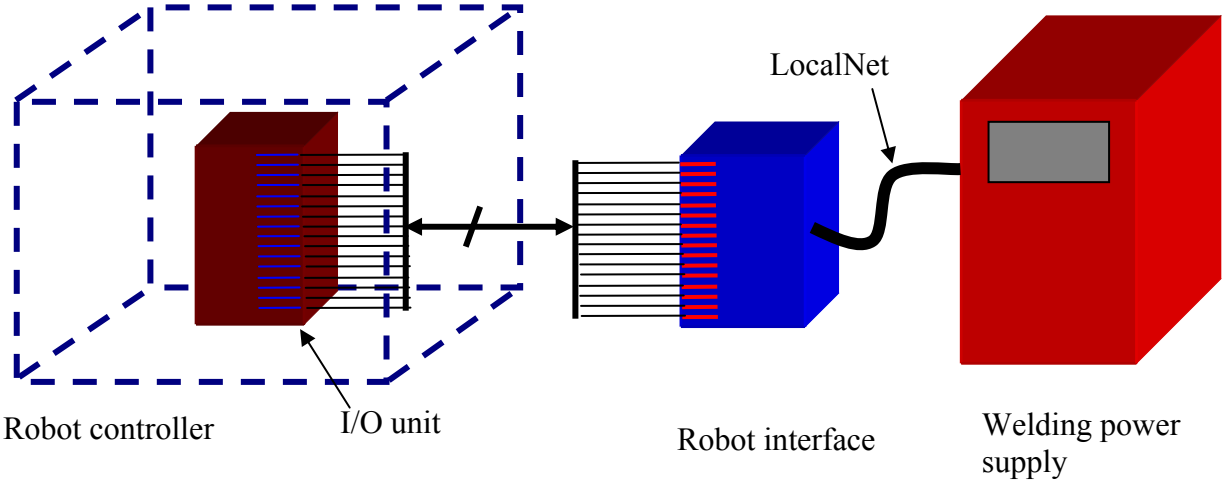


Figure 4.2.2-3. Communications between the robot and the power supply.

4.2.3 Camera Calibration Window

In the camera calibration window the camera parameters obtained in the last camera calibration sequence run are shown. This window opens every time the user clicks on the "*Show Calibration*" button or automatically after every camera calibration sequence. This window is shown in Figure 4.2.3-1. The rotation matrix, the translation vector and the focal point are shown. The focal point is expressed in pixels, therefore different focal points are specified for x and y since the scaling factor μ is not exactly the same for both axes.

To the right of the screen the error factors are shown for both axes, according to the camera calibration algorithm mentioned in previous sections.

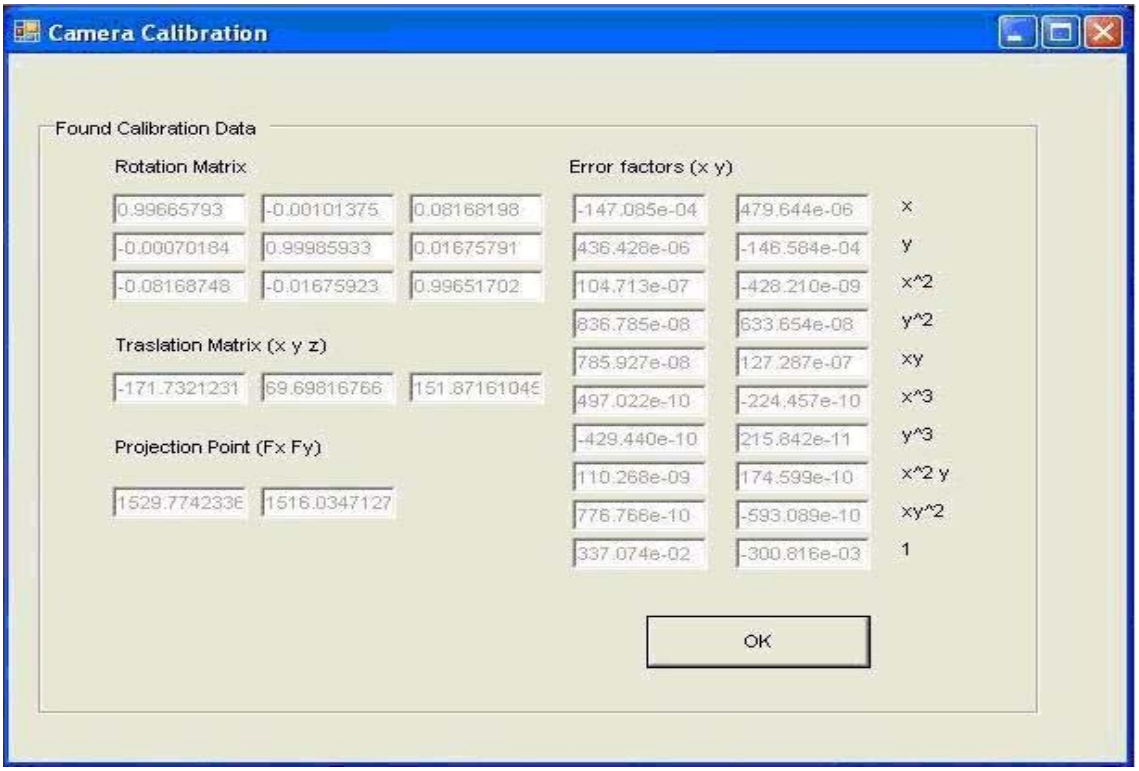


Figure 4.2.3-1. Camera calibration window.

By clicking the "*OK*" button, the window is closed.

4.2.4 Calculate Coordinates Window

This window has two presentation modes, depending on the button the user presses. If the user clicks on the "*Calculate Coordinates*" button, the window will have the appearance shown in Figure 4.2.4-1. In this mode the screen will have three buttons: "*Calculate Points*", "*Open Coordinates File*" and "*Close*". The "*Calculate Points*" button starts the 3D coordinate extraction sequence of the artificial characteristic points. The "*Open Coordinates File*" button was included to make it possible to load a set of coordinates for every point from a text file. This functionality allows the software to show calculation errors by

comparing the calculated coordinates and the loaded coordinates. This feature is only useful for testing the accuracy of the system. The other button, "*Close*", closes the screen. The text boxes in this mode are not populated.

In the second mode the window opens as shown in Figure 4.2.4-2. In this mode, the screen has additional buttons available and the text boxes are populated with information about the last extracted artificial feature points. The window takes this form after the user clicks on the "*Show Point*" button at the main screen or after the coordinate calculation sequence finishes.

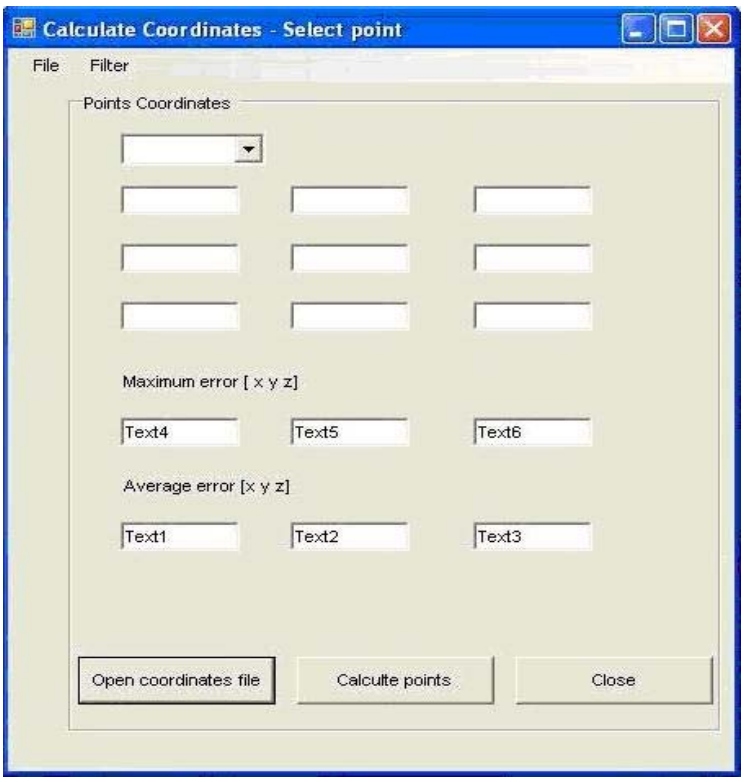


Figure 4.2.4-1. Calculate coordinates window (first mode).

The six lower text boxes are populated with the maximum and average absolute error in the three Cartesian directions. The nine upper textboxes are populated

with the real value, if a coordinate file was loaded, the estimated value and the error for *x*, *y* and *z* axes. These values are shown for the characteristic point selected by the user by using the dropdown menu.

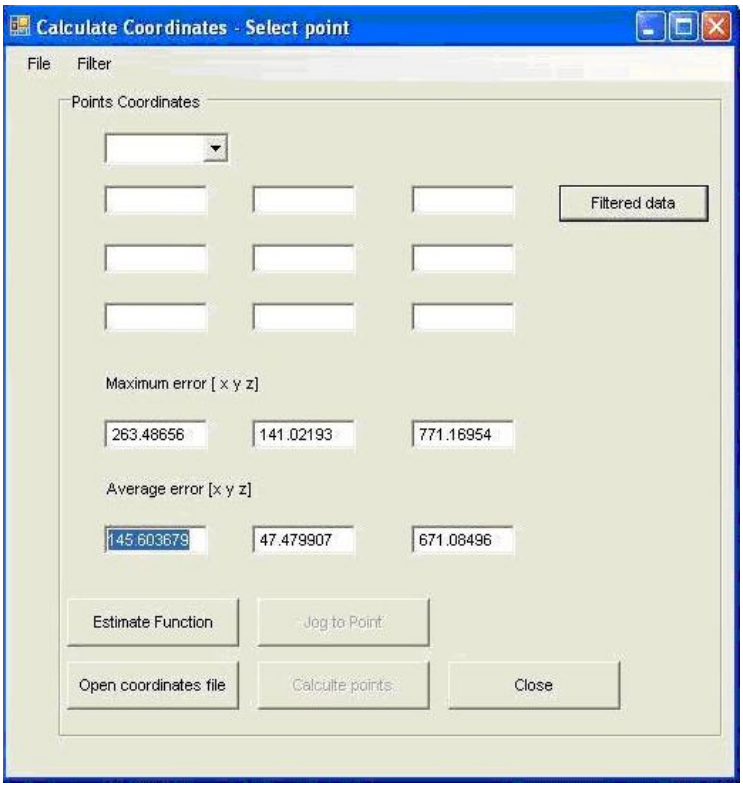


Figure 4.2.4-2. Calculate coordinates window (second mode).

A button designated "*Filtered data*" will also be available in this second screen. This button will show the *z* value after filtering for the point selected, the absolute maximum and the average. This information is shown in individual text boxes which become visible as soon as the button is pressed.

The "*Estimate Function*" button also becomes available in this second screen. This button opens a new form to estimate the function of the surface. This new window will be explained in the following section.

The main menu is standard for the two modes of operation and its options are shown in Figure 4.2.4-3. The first sub menu is "*File*" which only has one item, "*Close*", used to close the screen. The second sub menu is "*Filter*" which is used to modify parameters of the filter to be used by the system. The first option, "*Bandwidth*", allows the user to change the bandwidth of the filter. The second option changes the type of the filter, a circular or a squared FIR filter can be selected. In Figure 4.2.4-4, squared and a circular filter plots are shown in the frequency domain.

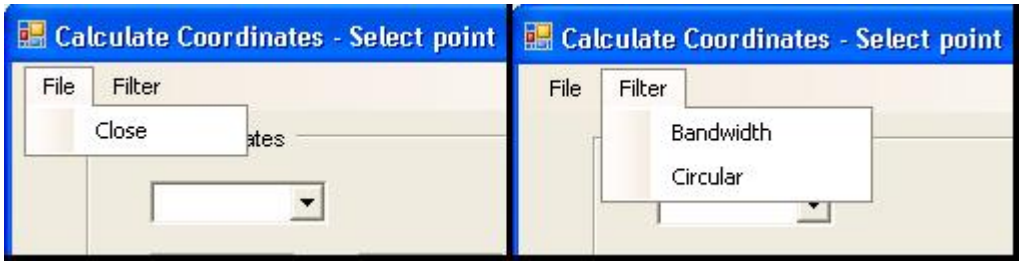


Figure 4.2.4-3. Camera calibration window menu.

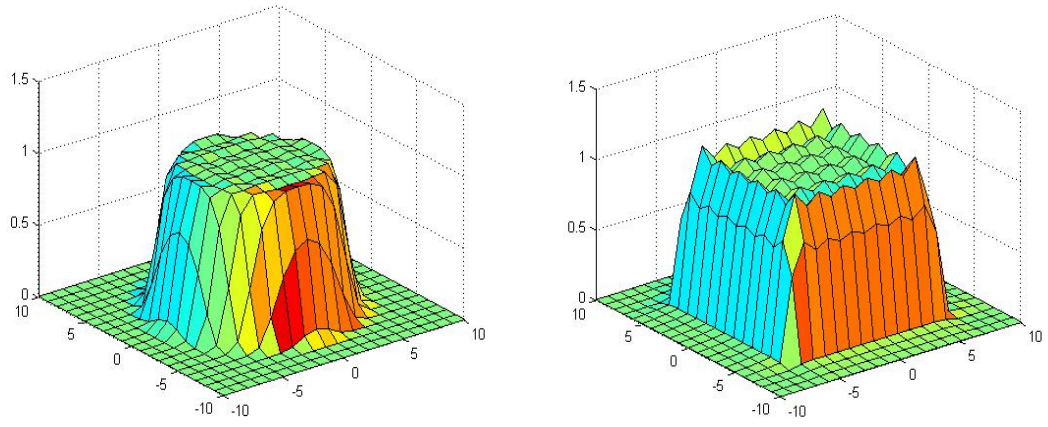


Figure 4.2.4-4. Circular and squared FIR filter plots in spatial frequency domain.

4.2.5 Estimate Function Window

This window allows the user to instruct the system to estimate a function that best describes the surface. This window is shown in Figure 4.2.5-1. Most of the options in this window were made for development purposes, therefore only the most relevant features of this window will be described.

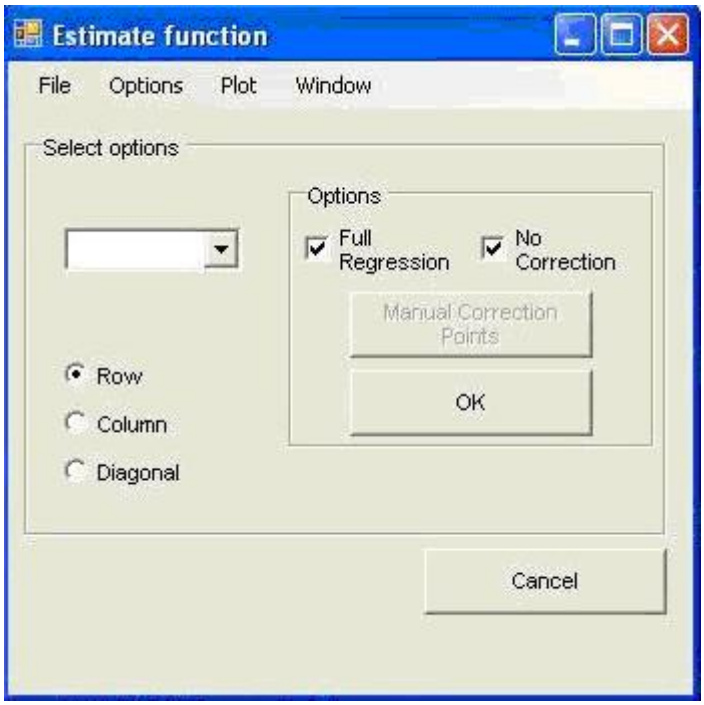


Figure 4.2.5-1. Estimate function window.

To obtain the function approximating the target surface, the "*OK*" button must be pressed. After this button is pressed, the screen expands to show the equation describing the surface as shown in Figure 4.2.5-2. The controls under the "*Regression*" control group can be hidden again by clicking on the "X" button. The resultant equation is displayed in the text box. If the "*No correction*" checkbox is not selected, the system will run a touch sensing routine for selected points unless a manual correction option has been selected by pressing the "*Manual correction points"* button. Plots of *x* vs. *y*, *x* vs. *z* and *y* vs.

z of any row, column or diagonal of the projected dot laser matrix can be generated according to the user selection by means of the radio buttons and the drop down menu. If the "*Full Regression*" Radio button is not pressed, the regression will be performed only taking into account the row, column or diagonal selected by the user, this feature was used for testing purposes of the system.

The menu options of this window are shown in Figure 4.2.5-3. Under the sub-menu "*File*", the user can save the *z* coordinate data of every point before and after regression in a text file. The "*Surface Matching*" option under the "*Options*" sub-menu will instruct the system to execute a third order regression if unchecked, otherwise the process described in previous sections to determine the regression order will be used.

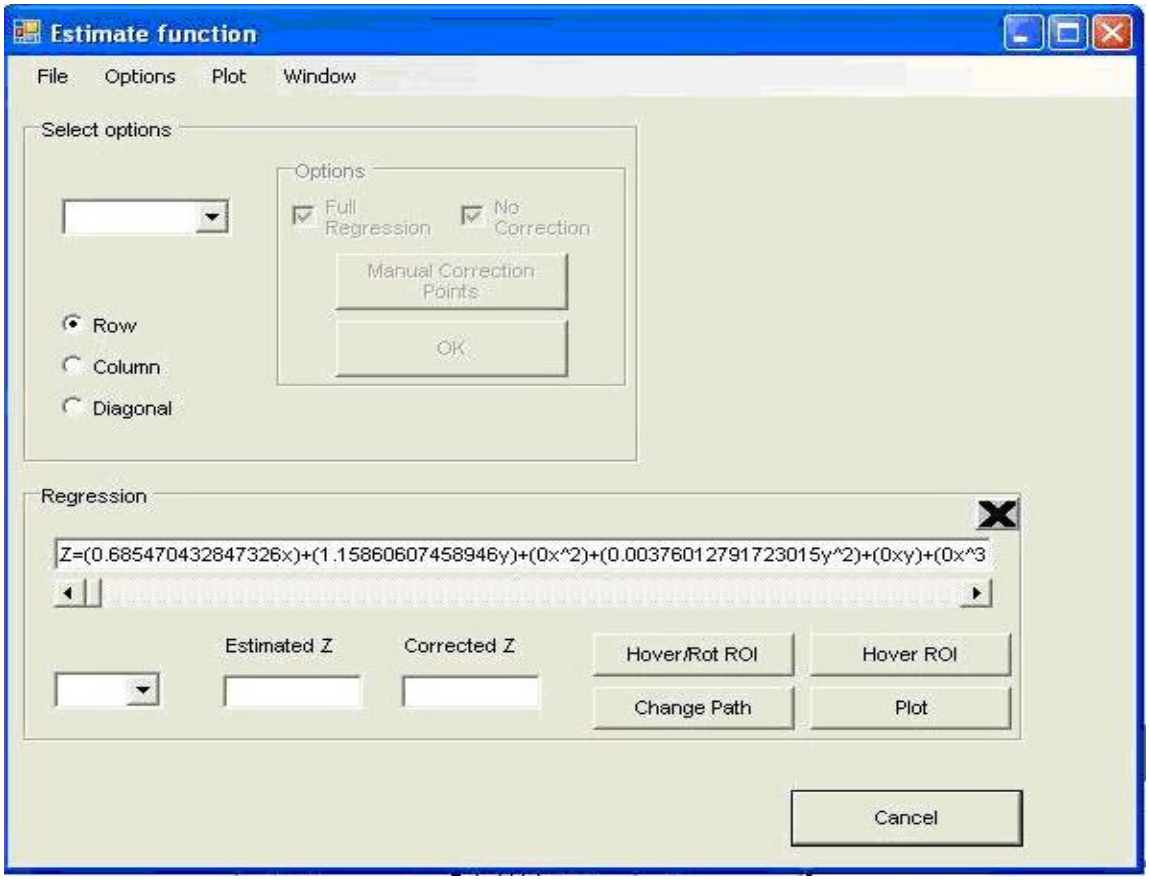


Figure 4.2.5-2. Estimate function window (expanded).

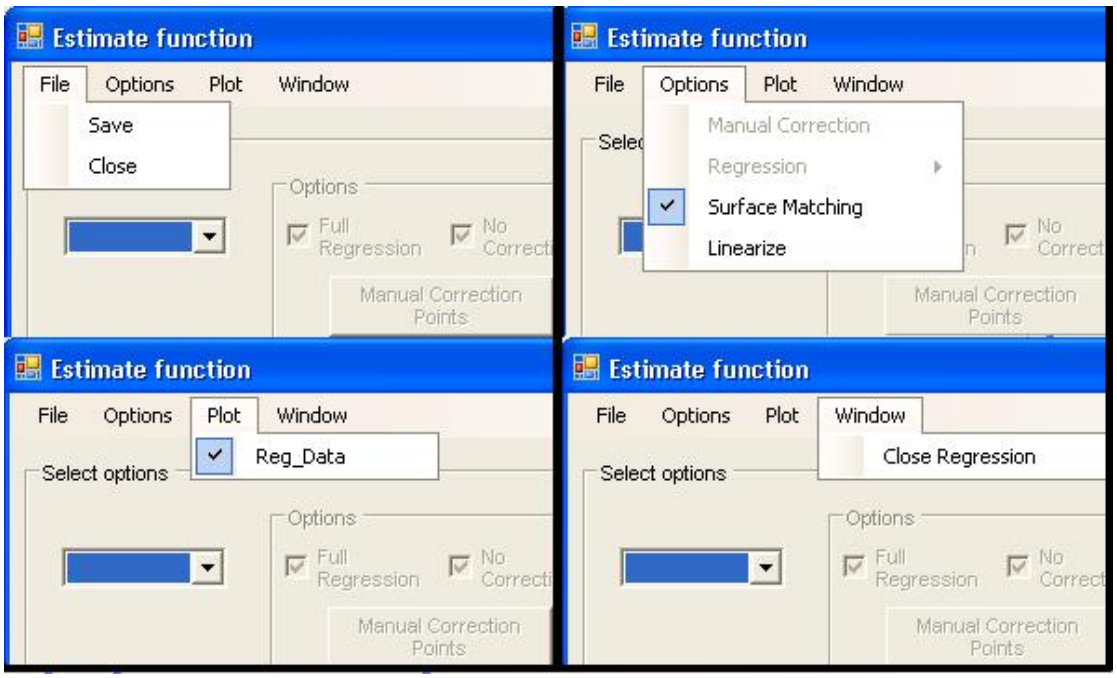


Figure 4.2.5-3. Estimate function window menus.

4.2.6 Toolboxes

Two small windows provide increased control and information concerning the different components of the system to the user. These windows consist of a camera set up toolbox and a robot status toolbox.

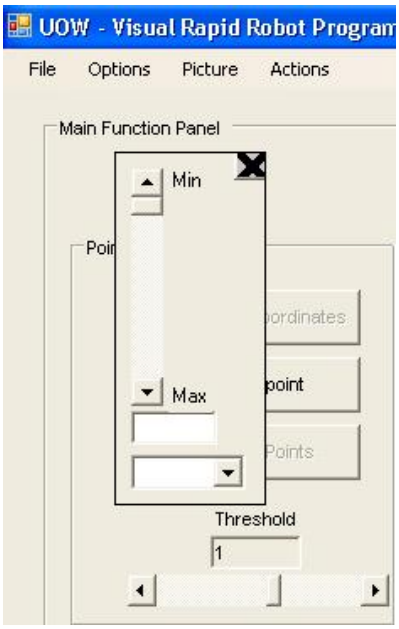


Figure 4.2.6-1. Camera setup toolbox.

The camera set up toolbox is shown in Figure 4.2.6-1. By using this toolbox, the user can adjust the brightness, gain, shutter speed and white balance (in U and V plane) of the camera by selecting a parameter in the dropdown menu and changing the value using the scroll bar. This toolbox can be launched by right clicking on the "*Camera Control*" box in the main window; it is available only when a connection has been established with the camera.

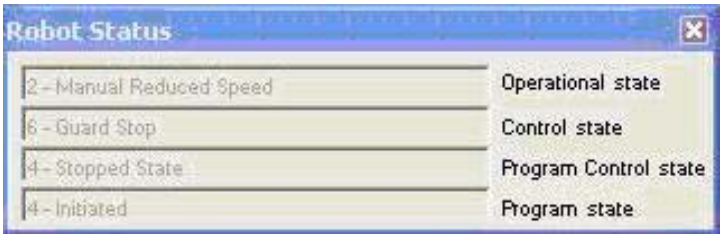


Figure 4.2.6-1. Robot status toolbox.

The robot setup toolbox shows the value of the status variables of the robot controller, these variables are: Operational State, Control State, Program Control State and Program State. This toolbox can be launched by right clicking on the "*Robot Commands*" box in the main window.

4.2.7 Automatic Sequence

In spite of the comprehensive control functions built into the system for development purposes, the process of generating a robot program using the proposed system is quite simple and the user interface is easy to use. The first step is to jog the robot to a position where the damage is visible and to ensure all laser points are visible by the camera. The user can then use the software application to take a picture of the surface to be repaired and indicate the perimeter of the area to be welded as shown in Figure 4.2.7-1.

Next, the user selects the appropriate configuration options included in the sub-menu "*Options*". The items "*Laser dots*" and "*AutoMode*" must be selected and it is advisable to select "*2Image-extraction*" (this option is checked by default). The user should verify the number of dots selected is correct. The "*Weld on*" option must be checked if a program with welding options is needed to be

generated. A typical configuration for automatic mode is shown in Figure 4.2.7-2. Following this the user will need to configure the welding parameters by using the *Welding Parameters* screen previously described in this chapter. After the welding supply parameters have been setup, the user clicks on "*Generate Lines"* under the "*Actions"* sub menu and the sequence starts.

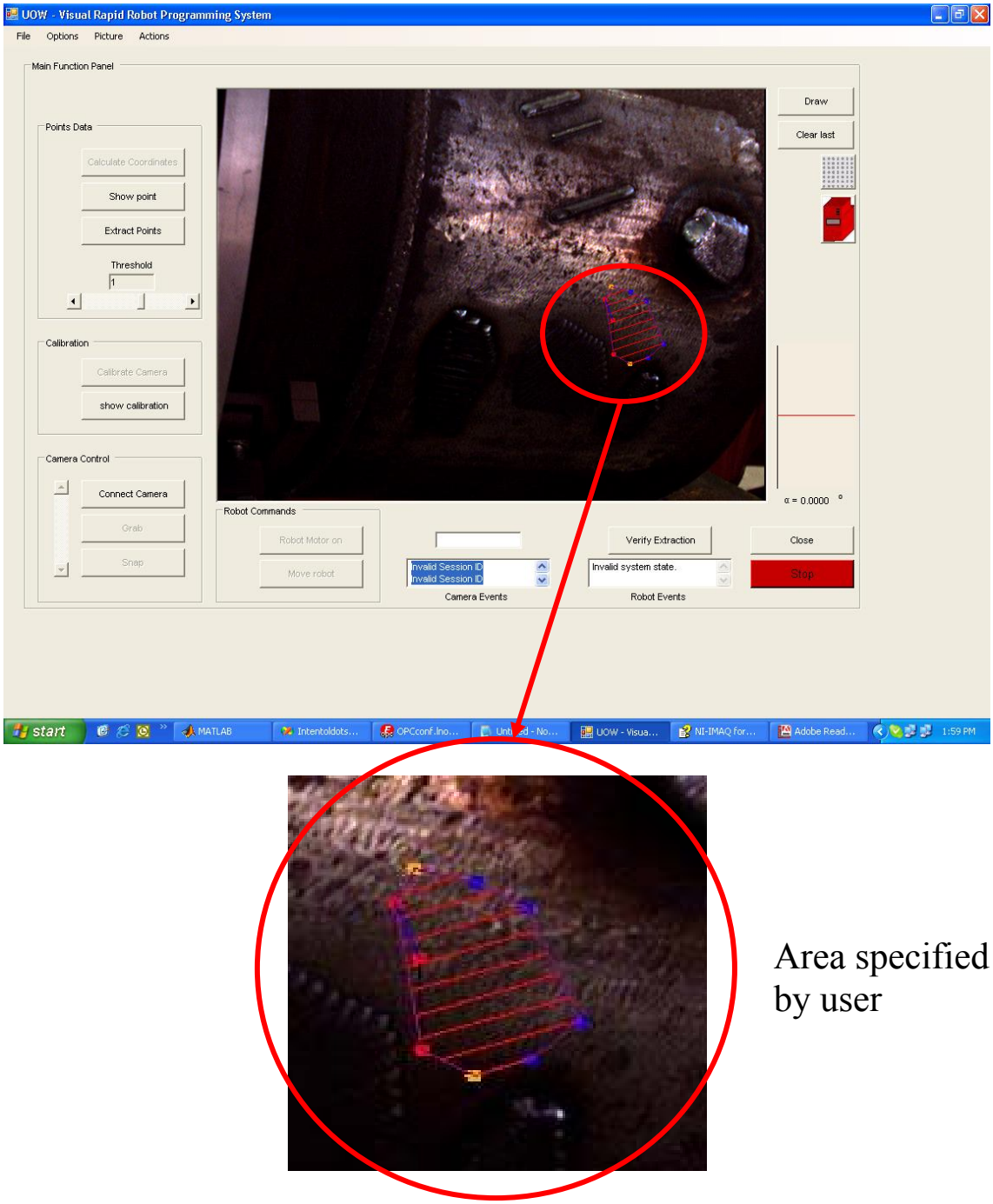


Figure 4.2.7-1. Main window with area specified by user.

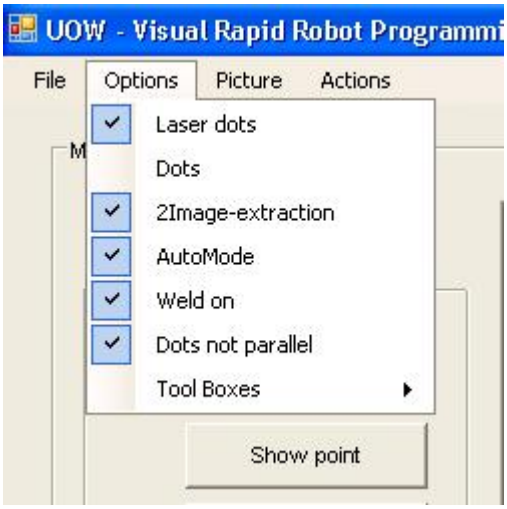


Figure 4.2.7-2. Typical configuration of "Options" menu for automatic mode operation.

The first system action is obtaining two shots of the artificial feature points in order to estimate their 3D coordinates, an example of two shots taken by the system during a sequence is shown in Figure 4.2.7-3. The extracted laser dots are represented by blue dots superimposed by the software to indicate the laser dots detected, in this way the user can monitor if there has been a bad detection or missing points.

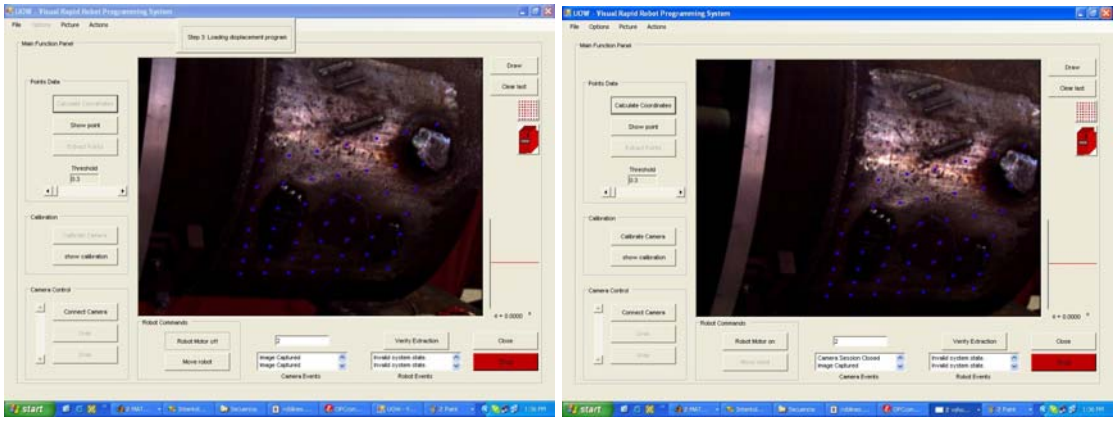


Figure 4.2.7-3. Extracted dots from the two shots during a coordinate calculation sequence.

The software indicates to the user the action being undertaken by the system by means of an information window located at the top centre of the computer screen. After the points have been extracted, the system performs the mathematical calculations to obtain the 3D coordinates and the equation that approximates the surface. After this is done, a window is opened showing a 2D plot of the welding lines as described in previous sections. Figure 4.2.7-4 shows a typical plot as it appears to the user for the area displayed in Figure 4.2.7-1 and the points extracted shown in Figure 4.2.7-3 for a distance of 10mm between weld beads.

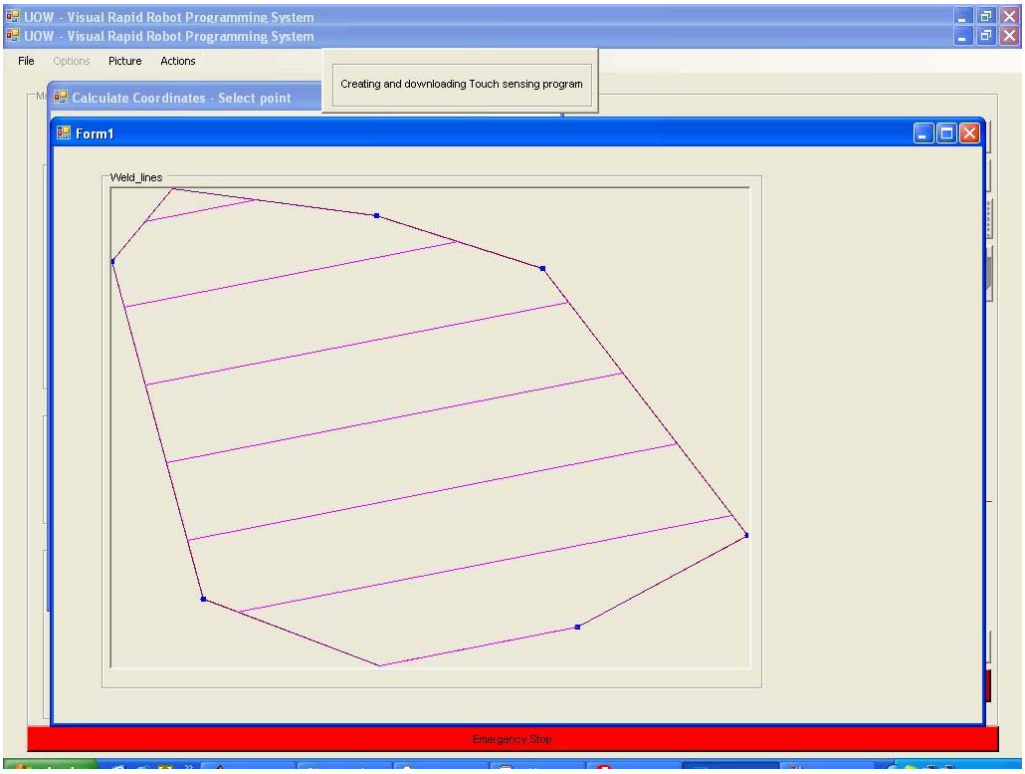


Figure 4.2.7-4. Two dimensional surface area plot.

Following this, a robot program is generated and downloaded to the robot controller to extract the corrected *z* distance of selected points by means of the touch sensing capability of the power supply. As indicated in Figure 4.2.7-4, progress of this action is notified to the user by means of the information window. When the touch sensing routine has finished execution, the coordinates of the sensed points can be transferred from the robot controller to the computer, therefore the estimated point coordinates can be corrected and a new equation for the surface can be obtained. It is possible then to generate the robot welding program and download it to the robot controller.

5. EXPERIMENT

5.1 EXPERIMENTAL METHODOLOGY

To test the camera calibration and the coordinate estimation method, a grid composed of black dots on a white background was used; the grid contained 64 dots (8 by 8). This grid was used to calibrate the camera and to test the accuracy of the system. As the camera is mounted on the welding gun, by moving the robotic arm it is possible to position the camera at different relative distances to the grid. In this way it is possible to estimate the coordinates of each dot in the grid from different camera positions. From each camera position, the coordinates for each one of the 64 points is extracted, the error between the real coordinates and the estimated coordinates is calculated for each dot, these errors are averaged and the maximum error is also found. The 3D coordinates of the points (real coordinates) are known and stored in a plain text file, this file is loaded into the system before the coordinate estimation is done. In general, this procedure was repeated for different positions ranging from 350mm to 900mm distance of the tool tip to the grid.

The average errors obtained for every camera position are averaged to obtain a global average error and maximum error. Three different methods to extract the coordinates were evaluated and the error obtained with each one of them is calculated.

Initially, the 64 dot calibration grid used was obtained by printing a pattern of black dots on white paper. The template was drawn using a CAD program to achieve as much accuracy as possible. In a latter stage, the pattern was replaced by an aluminium plate with holes drilled with high precision; this was placed on a black background. Figure 5.1-1 shows both test grids.

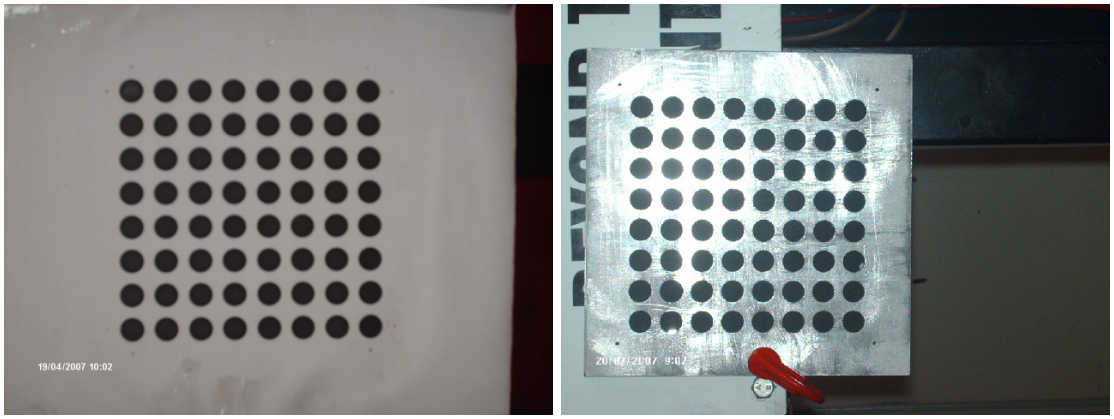


Figure 5.1-1. Paper and aluminium versions of the calibration grid.

Following this, a grid composed of red laser dots was used to test the system. To do so, it was necessary to adapt the extraction algorithm to be able to recognise the dots.

Initially National Instruments LabVIEW [70] was used to extract the feature points and as a tool for image analysis in further development stages. The high level blocks included in the Vision Assistant module of LabVIEW were used to develop an application to extract the dots, however limitations were found due to the change of the size and shape of the dots in the image depending of the camera position. Nevertheless, this software package was an important analysis tool during the project.

Another important engineering tool used was The Mathworks Matlab [71]. This program was used to analyse and process data and test important processing routines before they were coded in Microsoft Visual Basic. In the first stages of the project, the processing of the images was done in Matlab and Visual Basic was used as an interface between the user, the robot, and the camera. However, as the project progressed the whole interface and processing code was re-written in Microsoft Visual Basic 6.0 [72] and the National Instruments IEEE1394 Vision libraries were used to establish communications with the camera. Visual Basic 6.0 did not deliver a high performance relating to speed for image processing routines; however it proved to be adequate for the requirements of the project. Nevertheless, the program was finally migrated to Microsoft Visual Studio 2005 [73] providing improved performance.

5.2 ARTIFICIAL FEATURE POINTS

Structured laser light was selected as a means of generating the artificial feature points on the target object surface.

Initially it was proposed to use a device which generated a dot matrix pattern using parallel collimated laser beams. The objective was to achieve enough laser power so the laser light dots could be clearly detectable through the whole working range (around 1 metre) and would offer an acceptable immunity level to environmental light. According to the manufacturer of laser products Stockeryale [74] the approximate amount of power needed for such a distance is 3mW for an alignment application. The aim was to develop a device which

would also cost less than a commercial structured laser light module; by using multiple low power, low cost laser diodes.

A device was designed consisting of 64 10 mW laser diodes with their respective collimating lenses mounted in an aluminium base designed to hold the diodes parallel to each other in such a way that a diode matrix would be formed as shown in Figure 5.2-1 (A brief theory about lenses is found in appendix F). To test the concept, a prototype of the device was built for holding only four diodes (see Figure 5.2-5). The alignment of only four diodes proved to be difficult and in order to achieve the desired outcome it was necessary to significantly increase the complexity of the device to allow precise individual alignment of each diode. This meant a significant increase in cost and time, and for this reason this structured laser light device design was discarded.

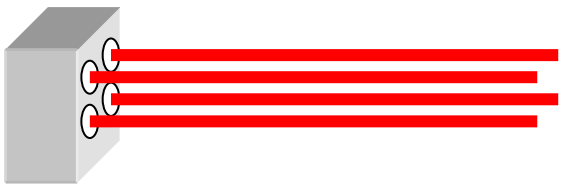


Figure 5.2-1. Parallel laserdiode beams matrix.

The alternate solution was to use a device which projected the laser light matrix using divergent laser beams (Figure 5.2-6). This kind of device is commercially available but at a considerable price for the amount of power required. Therefore a laser projector module was built using a diode laser, a collimating lens, a diffraction grid (seven by seven *Lasiris* projection head) and an aluminium holder assembly (Figure 5.2-7). The resultant laser projector device

is shown in Figure 5.2-2. A simple electronic circuit with overload protection was designed to drive the diode. The electronic circuit was placed into a plastic enclosure; a toggle switch was included for security reasons (refer to Appendix G for a brief review on Laser Safety) to provide electromechanical isolation from the power supply as well as a green LED to indicate if the switch is in the "ON" position, this enclosure is shown in Figure 5.2-3.

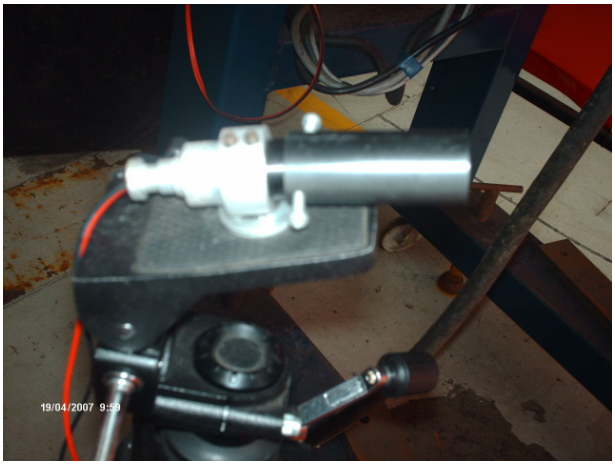


Figure 5.2-2. Laser projector device.

The circuit is powered by the robot controller power supply and the laser LED is turned on by means of a digital output of the robot controller. The schematic drawing of the driver circuit is shown in Figure 5.2-4.

Figure 5.2-3. Laser projector driver circuit enclosure.

The laser diode used was a Hitachi HL6535MG capable of an output up to 90mW of continuous optical power at a wavelength of 680nm. The laser LED has a 5.6mm TO18 case. This relatively high power diode was used because it was necessary to obtain an adequate intensity of the laser dots along the whole working range. Preliminary optical test were conducted with a laser pointer of 1mW and it was determined that 1mw gave sufficient power for one laser dot for the application; subsequent tests with a 10 mW diode and the diffraction matrix yielded poor results regarding optical intensity, indicating the need for of a more powerful laser diode. Since 49 laser dots are generated it was desirable to obtain a laser diode of at least 50mW. Although the maximum continuous power the Hitachi HL6535MG laser can deliver is considered high enough to achieve a good working range, it can be increased to 240mW if pulsed at a 50ns period with 40% duty cycle (if more output power is needed).

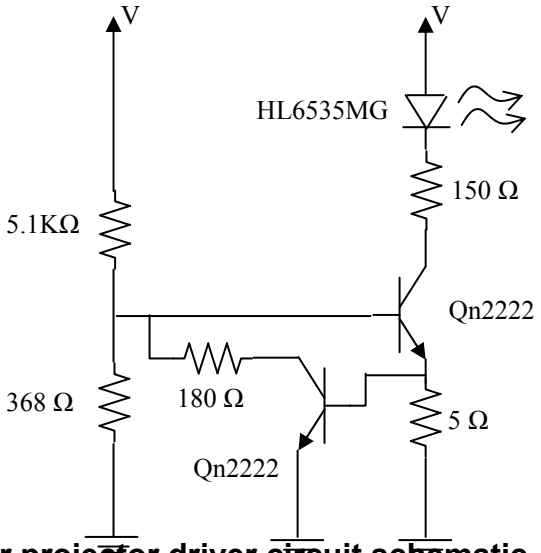


Figure 5.2-4. Laser projector driver circuit schematic.

A conventional digital projector was also tested as source of the characteristic points. A matrix of white dots with a black background was the template used to generate the projected pattern, as shown in Figure 5.2-8.

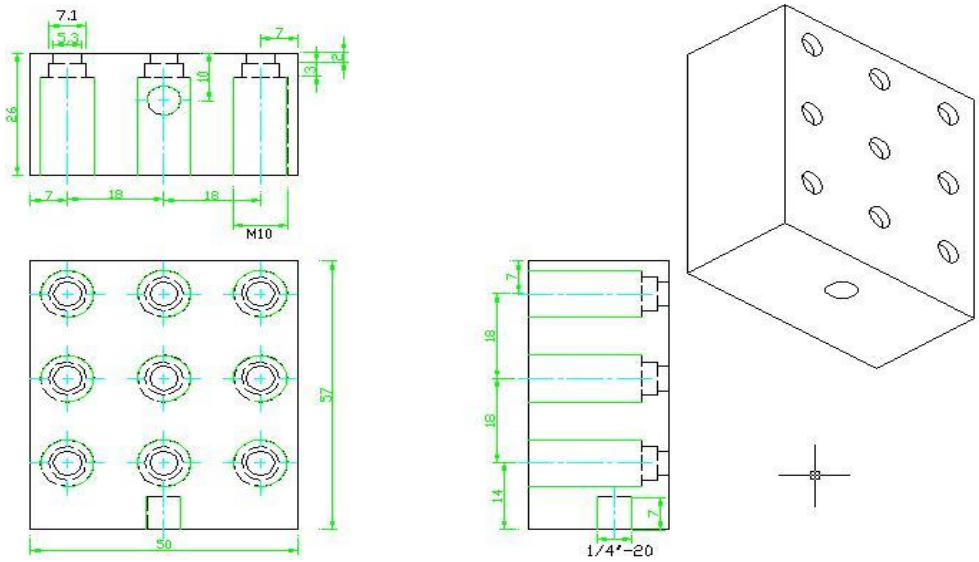


Figure 5.2-5. Parallel laser diode beams matrix assembly.

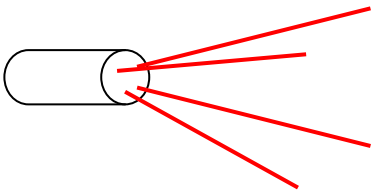


Figure 5.2-6. Divergent laser diode beams matrix.

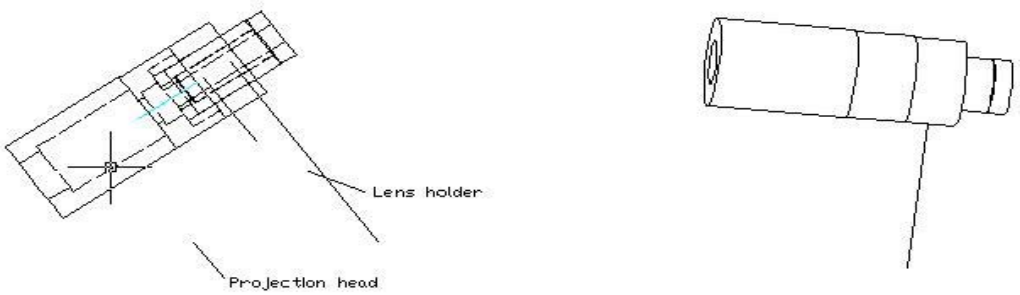


Figure 5.2-7. Laser diode assembly.

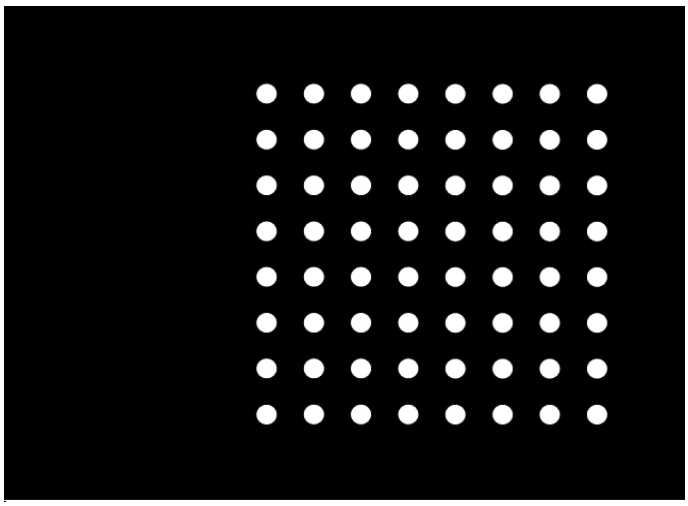


Figure 5.2-8. Pattern used in digital projector.

The projected pattern on an opaque flat surface is shown in Figure 5.2-9. The size of the pattern is of the order of 20 by 20 centimetres with the projector located at a short distance from the surface. This is due to the large angles of view of the projector; in order to produce smaller dots over a smaller area it is necessary to reduce the size of the template. It can be seen that the projected dots are sharp and easy to identify, therefore it was not difficult for the extraction algorithm to find the dots in the image. It was necessary however to modify the colour settings of the algorithm in order to recognise white dots

instead of the black dots of the test grid. However one of the bigger drawbacks of the projector is the portability (if compared with the laser projector), even though state of the art projectors are notably smaller than few years ago. Additionally, digital projectors need a source that provides them with the image information, such as a computer.

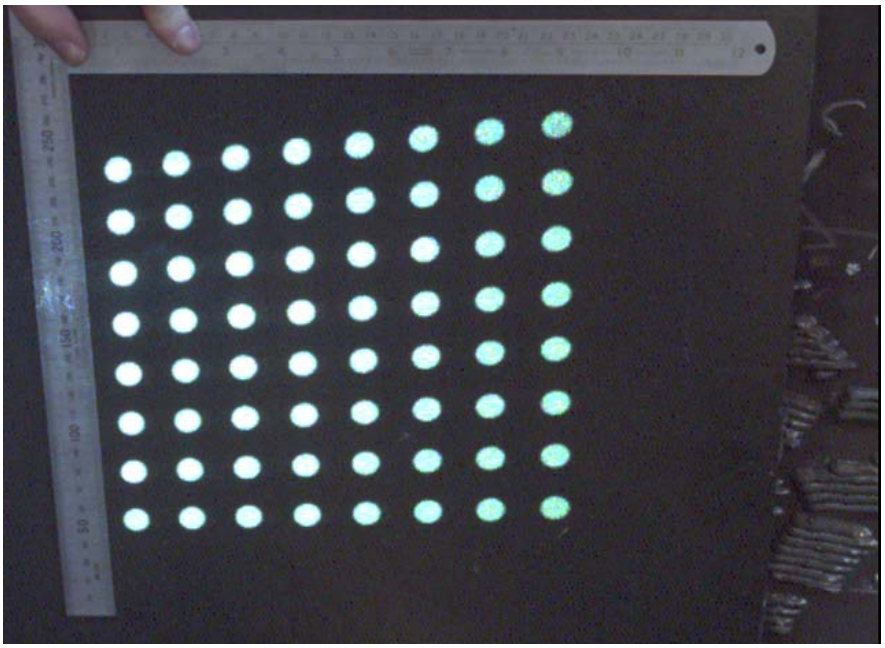


Figure 5.2-9. Pattern used in projector.

6. EXPERIMENTAL RESULTS

6.1 LENS DISTORTION

When using any vision systems, it is not unusual to find a distortion term included in the camera model. However, as previously mentioned, Sheng et al. [33] claims that when the optical distortion introduced by the camera lens is small, to include distortion factors in the camera model does not improve the accuracy and can even worsen it. This was the case in this work when trying to include distortion factors in the calibration equations of the camera. The method proposed by Motta et al. [31], which includes a radial distortion term, was used in order to test the applicability of distortion factors in the camera model; however little or no improvement was found as shown in Table 6.1-1; detailed tables of the the errors obtained with and without distorrion factors can be found in Appendix C. In this table the average and maximum errors are shown for a total of 2964 points extracted from different distances. Slight variations in the distortion factor calculation were tried in an attempt to get a noticeable improvement but this did not improve the situation.

In conclusion, after many trials it was found that the distortion introduced by the lens used was so small that it was not worth including a factor to compensate for it in the camera calibration model.

Table 6.1-1. Errors (mm) obtained with and without distortion factors in the calibration equation.

Average x	Average y	Average z	Max. x	Max. y	Max. z	
With distortion factor						
0.782601	0.557377	15.48044	5.3324	3.2057	63.1349	Total
0.562081	0.49869	12.62364	2.6233	2.3319	38.02	vect(1)
1.003121	0.616065	18.33724	5.3324	3.2057	63.1349	geo(2)
Without distortion factor						
0.786927	0.568436	15.78403	5.343	3.2662	63.4386	Total
0.567005	0.508668	12.93662	2.6283	2.3813	38.3726	Vect
1.006848	0.628203	18.63145	5.343	3.2662	63.4386	Geo

(1) Vectorial calculation
(2) Geometric calculation

6.2 CAMERA CALIBRATION METHOD

In order to evaluate the camera calibration techniques considered in this work, it was necessary to undertake several tests that accumulated a significant number of data points. Two basic methods were tested: the geometric method and the planes method. These two methods are described in Chapter 2.2.2. The modified planes calibration method explained in Chapter 3.2 was also evaluated. A summary table containing the results of the tests is presented below, more detailed results are presented in Appendix A.

Table 6.2-1. Errors (mm) obtained using different camera calibration methods and calculation techniques

Average x	Average y	Average z	Max. x	Max. y	Max. z	
Modified Planes (150x150)						
0.68303	0.570733	14.31871	5.2931	3.1971	69.1806	Total
1.206718	0.726668	23.29162	5.2931	3.1971	69.1806	Geo(1)
0.275918	0.394089	4.499204	1.1722	1.5055	19.2802	Vect-geo(2)
0.566454	0.591443	15.16531	2.4364	2.26795	42.65625	Vect(3)
Modified Planes (100x100)						
0.699334	0.582404	14.33724	5.3013	3.248	69.2136	Total
1.245886	0.742929	23.97442	5.3013	3.248	69.2136	Geo
0.287285	0.398009	3.870795	1.318438	1.5933	18.6245	Vect-geo
0.564832	0.606275	15.1665	2.4296	2.2851	42.6629	Vect
Planes(quadratic)						
1.041137	0.608448	19.16312	5.9225	3.6328	72.7351	Total
1.61095	0.761964	28.56981	5.9225	3.6328	72.7351	Geo
0.60635	0.410346	8.893346	1.8785	1.574	23.2384	Vect-geo
0.906111	0.653032	20.02621	3.1874	2.71375	46.04305	Vect
Geometric (Tsai)						
0.709543	0.526487	12.95156	5.343	3.2662	63.4386	Total
1.251971	0.698282	22.08939	5.343	3.2662	63.4386	Geo
0.293654	0.363654	3.349607	1.2602	1.3085	14.6186	Vect-geo
0.583004	0.517525	13.41568	2.62445	2.27465	36.8919	Vect
Planes (linear)						
0.933621	0.607541	21.42932	5.7163	3.5843	76.004	Total
1.479707	0.757068	30.57783	5.7163	3.5843	76.004	Geo
0.485611	0.405114	10.7959	1.6482	1.5688	26.6305	Vect-geo
0.786221	0.643325	21.98937	2.96705	2.68955	49.6184	Vect

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

The results shown in Table 6.2-1 are the errors obtained in millimetres of the estimated 3D coordinates of 64 points located on a plane. Twelve extractions

were performed, each from a different position, summing up 768 points. The test was repeated, thereby obtaining a total of 1536 points. In total, four calibration options are included in the table. However, the modified planes method was repeated, with the number of points used to obtain the projection focal point increased from 100x100 to 150x150. The remaining calibration option was the planes method but using a linear polynomial instead of quadratic. Each calibration method was tested using three different calculation techniques; each calculation technique has been previously explained. The vectorial-geometric technique combines the results from the other two methods.

According to the results presented above, it can be seen that the planes calibration method yields similar result in the *y* coordinate compared to the modified planes method. However, the results in the *x* and *z* coordinates are improved in the modified planes method. On the other hand, the modified planes calibration method yields similar results to the geometric method although the geometric method seems to perform slightly better in *z* direction. It can be seen that no improvement is obtained when increasing the number of projection points used for the planes method.

In the combined calculation method only the *x* information was considered for the geometrical component. Therefore the errors were recalculated using only the data supplied by the displacement in *x* direction, the results are given in Table 6.2-2.

According to the results in Table 6.2-2, it can be seen that the error behaviour relative to the camera calibration technique shows a similar behaviour to that found when *x* and *y* data are considered. However, the linear planes method gives a noticeable improvement in the error in *x* coordinate relative to the other methods.

Table 6.2-2. Errors (mm) obtained using different camera calibration methods and calculation techniques and only *x*-axis data.

Average x	Average y	Average z	Max. X	Max. y	Max. z	
Modified Planes (150x150)						plan(fo)
0.282089	0.421223	3.951954	1.7637	2.0176	17.0689	Total
0.279718	0.464443	3.918279	1.6917	2.0176	16.8505	Geo(1)
0.283054	0.381336	3.952836	1.7277	1.504	16.8625	vect-geo(2)
0.283496	0.417889	3.984746	1.7637	1.3066	17.0689	Vect(3)
Modified Planes (100x100)						plan(f)
0.287933	0.431276	3.95058	1.7567	2.0305	17.0722	Total
0.285657	0.477629	3.916386	1.6831	2.0305	16.8593	Geo
0.289189	0.381064	3.951411	1.7199	1.5195	16.8712	vect-geo
0.288954	0.435136	3.983943	1.7567	1.3585	17.0722	Vect
Planes(quadratic)						plan(squ)
0.354318	0.405437	5.695371	1.2892	1.4726	17.9715	Total
0.358382	0.396829	5.729436	1.2889	1.4726	17.9715	Geo
0.354286	0.343186	5.696311	1.2886	1.2805	17.7787	vect-geo
0.350286	0.476296	5.660368	1.2892	1.4127	17.5859	Vect
Geometric (Tsai)						Geom.
0.261357	0.401969	4.037398	1.6131	1.8703	19.7832	Total
0.260121	0.442439	3.97825	1.5431	1.8703	19.7602	Geo
0.261643	0.429489	4.035043	1.5781	1.4293	19.7371	vect-geo
0.262307	0.333979	4.0989	1.6131	1.1312	19.7832	Vect
Planes (linear)						plan(lin)
0.285573	0.404243	7.137211	1.2139	1.5202	20.4297	Total
0.289111	0.403661	7.233636	1.1485	1.5202	20.4297	Geo
0.285768	0.341593	7.180325	1.1776	1.2775	20.2356	vect-geo

0.282729	0.466061	7.122711	1.2139	1.41	20.0414	Vect

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

The planes method calibration was carried out based on the approach proposed by Gremban et al. [36] (explained above in Section 6.3). However, Equation 25 has an implicit approximation which affects accuracy that is explained below. According to Equations 9 and 10 and recalling that **R=[$\mathbf{R_1}$; $\mathbf{R_2}$; $\mathbf{R_3}$]** the following is obtained:

$$x_i = \mu_x f \frac{\mathbf{R}_1 \bullet \mathbf{P} + t_x}{\mathbf{R}_3 \bullet \mathbf{P} + t_z} = \mu_x f \frac{\mathbf{R}_1 \bullet \mathbf{P} - \mathbf{R}_1 \bullet \mathbf{T}'}{\mathbf{R}_3 \bullet \mathbf{P} - \mathbf{R}_3 \bullet \mathbf{T}'}$$

[Equation 114]

$$x_i = \mu_x f \frac{\mathbf{R}_1 \bullet (\mathbf{P} - \mathbf{T}')}{\mathbf{R}_3 \bullet (\mathbf{P} - \mathbf{T}')} = f_x \frac{\mathbf{R}_1 \bullet \frac{(\mathbf{P} - \mathbf{T}')}{\|\mathbf{P} - \mathbf{T}'\|}}{\mathbf{R}_3 \bullet \frac{(\mathbf{P} - \mathbf{T}')}{\|\mathbf{P} - \mathbf{T}'\|}} = f_x \frac{\mathbf{R}_1 \bullet \mathbf{v}_n}{\mathbf{R}_3 \bullet \mathbf{v}_n}$$

[Equation 115]

The vector **T'** is the translation vector in world coordinates whereas t_x and t_z are the translation vector components given in camera coordinates. It can be seen that Equation 115 is similar to Equation 25, except for the dividing factor $\mathbf{R}_3 \bullet \mathbf{v}_n$ which is approximated to 1 in Equation 25. In fact, in practice this dot product is close to unity unless the camera in use has a wide angle of view. For example the camera used in this project has an approximate angle of view of 28^o in *x* and is less in *y*, therefore the dot product under consideration would be between 0.97 and 1, i.e. between cos(14) and cos(0), since the two vectors

have a magnitude equal to 1. The larger the focal length of the camera lens in use, the smaller the angle of view and the more valid the approximation in Equation 25.

If after calculating the rotation matrix using Equation 25, f_x and f_y are recalculated by applying Equation 115 or by finding these values using the projection equation for *N* points and averaging theses values, the errors in Table 6.2-3 are obtained.

The results in Table 6.2-3 show a similar result for all the calibration methods except for the linear planes. This method presents the highest errors in *x* and *z* coordinates.

The results in Table 6.2-3 coincide with the findings of Champleboux et al. [37] who claim to have obtained better results with the quadratic planes method than with the geometric method and that the linear planes method had a lower performance than these two.

Table 6.2-3. Errors (mm) obtained using different camera calibration methods and calculation techniques using only *x*-axis data. (f_x and f_y refined for planes methods).

Average x	Average y	Average z	Max. X	Max. y	Max. Z	
Modified Planes (150x150)						plan(fo)
0.282526	0.418814	3.934435	1.7637	2.0176	17.0689	total
0.281088	0.4549	3.863027	1.6917	2.0176	16.8505	geo(1)
0.283054	0.381336	3.952836	1.7277	1.504	16.8625	vect-geo(2)
0.283496	0.417889	3.984746	1.7637	1.3066	17.0689	vect(3)
Modified Planes (100x100)						plan(f)
0.287933	0.431276	3.95058	1.7567	2.0305	17.0722	total
0.285657	0.477629	3.916386	1.6831	2.0305	16.8593	geo
0.289189	0.381064	3.951411	1.7199	1.5195	16.8712	vect-geo
0.288954	0.435136	3.983943	1.7567	1.3585	17.0722	vect
Planes(quadratic)						plan(squ)
0.256829	0.393825	3.971693	1.3281	1.6216	15.0228	total
0.259139	0.405204	3.932893	1.303	1.6216	14.9841	geo
0.256386	0.399596	3.974575	1.3012	1.4027	14.9455	vect-geo
0.254961	0.376675	4.007611	1.3281	1.3378	15.0228	vect
Geometric (Tsai)						geom
0.261357	0.401969	4.037398	1.6131	1.8703	19.7832	total
0.260121	0.442439	3.97825	1.5431	1.8703	19.7602	geo
0.261643	0.429489	4.035043	1.5781	1.4293	19.7371	vect-geo
0.262307	0.333979	4.0989	1.6131	1.1312	19.7832	vect
Planes (linear)						plan(lin)
0.362793	0.413693	4.267573	1.7151	2.0383	20.2435	total
0.359232	0.444054	4.157293	1.6403	2.0383	20.2367	geo
0.360639	0.387868	4.226796	1.6777	1.5379	20.2299	vect-geo
0.362907	0.399164	4.299229	1.7151	1.5164	20.2435	vect

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

6.3 POINT CALCULATION METHOD

As explained in Chapter 3, three methods were tested for calculation of the coordinates of the points. These methods are the geometrical, the vectorial and a combination of both. The errors obtained with each of these methods are summarised in Table 6.3-1. This table presents the average and maximum values from Table 6.2-1 for each calculation method. It can be seen that the combined technique (vectorial-geometric) presents considerable improvements compared to the other two methods.

Table 6.3-1. Maximum and average errors (mm) obtained for different point calculation methods.

Average x	Average y	Average z	Max. x	Max. y	Max. z	
1.359046	0.737382	25.70061	5.9225	3.6328	76.004	geo(1)
0.389763	0.394242	6.28177	1.8785	1.5933	26.6305	vect-geo(2)
0.681324	0.60232	17.15261	3.1874	2.71375	49.6184	vect(3)

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

According to Table 6.3-1 in every aspect the combined method outperforms the other two, the average and maximum error values for the three coordinates (*x*, *y* and *z*) are the smallest for this method. However, as already mentioned, for the geometrical approach the information given by the two axes (*x* and *y*) were taken into account, while for the combined method only the contribution of the *x* coordinate was considered to obtain the geometrical component.

It was found that using only the information supplied by the displacement in the *x* axis, the results improve notably. The total average and maximum errors for the different calculation methods are summarised in Table 6.3-2 when only the information in the *x* axis is considered.

Table 6.3-2. Maximum and average errors (mm) obtained for different point calculation methods using only *x*-axis data.

Average x	Average y	Average z	Max. x	Max. y	Max. z	
0.294598	0.437	4.955197	1.47106	1.78224	18.37424	geo(1)
0.294788	0.375334	4.963185	1.49838	1.40216	18.29702	vect-geo(2)
0.293554	0.425872	4.970134	1.52732	1.3238	18.31032	vect(3)

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

To take the information in only one axis for the geometrical vector, the projection of the vectors over the *x*-*z* plane is considered. It is believed that this behaviour is due to the fact that the amplitude of the displacement in pixels in the *x* coordinate is about 1.3 times the displacement in *y*. Therefore, the information in the *x* coordinate is, in proportion, less influenced by noise and calculation errors since the error introduced by these factors should be of the same magnitude in both axes. In the earlier stages of the research, it was found that displacements smaller than 20 pixels produced a considerable error. With the two defined relative positions for the camera to take the shots, the displacement obtained is between 86 and 40 pixels for *x* and between 63 and 27 pixels for *y*.

When the focal distances in pixels (f_x and f_y) are recalculated according to Equation 115 for the planes methods, the results obtained are those given in Table 6.3-3.

Table 6.3-3. Maximum and average errors (mm) obtained for different point calculation methods using only *x*-axis data (f_x and f_y refined for planes methods).

Average x	Average y	Average z	Max. x	Max. y	Max. z	
0.289048	0.444845	3.96957	1.57224	1.91566	17.73816	geo(1)
0.290182	0.395871	4.028132	1.60092	1.47868	17.72924	vect-geo(2)
0.290525	0.392569	4.074886	1.63534	1.3301	17.83812	vect(3)

(1) Vectorial calculation
(2) Vectorial-Geometric calculation
(3) Geometric calculation

It can be seen from the results in the table above that the errors obtained with the different calculation methods are very similar. The geometric method yields a slightly higher error in the *y* coordinate.

6.4 PROGRAM GENERATION

The robot program was generated using the RAPID language syntax, which is native to the ABB robot control unit. The code is created into a text file with ".prg" extension and downloaded to the robot via an Ethernet link. The creation of the text file is undertaken by the application developed here in Visual Basic. As previously described, the established weld paths are divided into small segments to produce an approximated shape of the surface as shown in Figure

6.4-1. A weld instruction is created between each line segment to create the full weld line.

Using this approach it was possible to create the program in just a few minutes; from the moment the user has selected the weld repair area to the moment the program is fully loaded into the robot controller. The major time consuming stage of the process is the refinement of the coordinate, i.e. when the robot is instructed to move to selected points and retrieve the exact *z* information for these points using the touch sensing capability of the power supply.

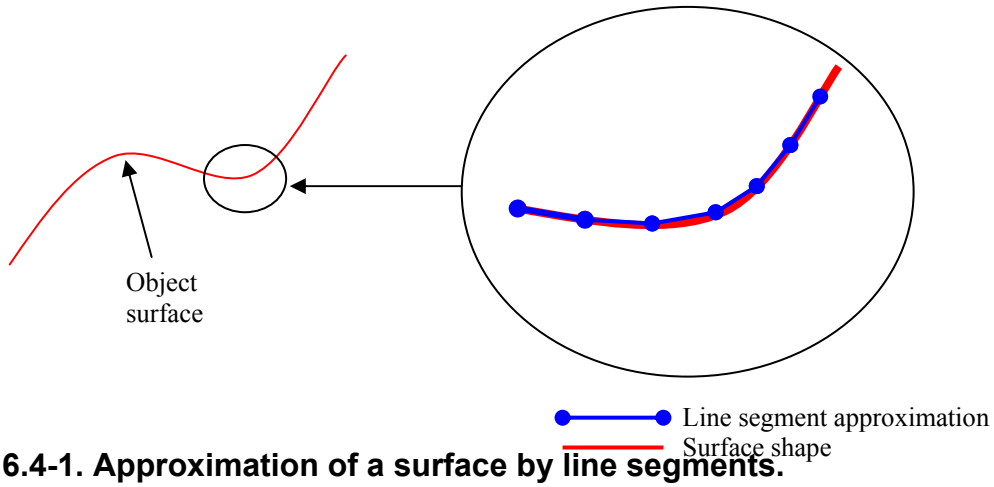


Figure 6.4-1. Approximation of a surface by line segments.

6.5 WELDING TRIALS (VALIDATION)

In order to test the system, several different welding trials were performed on curved surfaces placed at unknown world coordinates. All welding trials were performed using the GMAW welding power supply in CMT mode. A pipe, shown in Figure 6.4-1, was initially used as the target. The final weld results are shown in Figures 6.5-2 to 6.5-5.

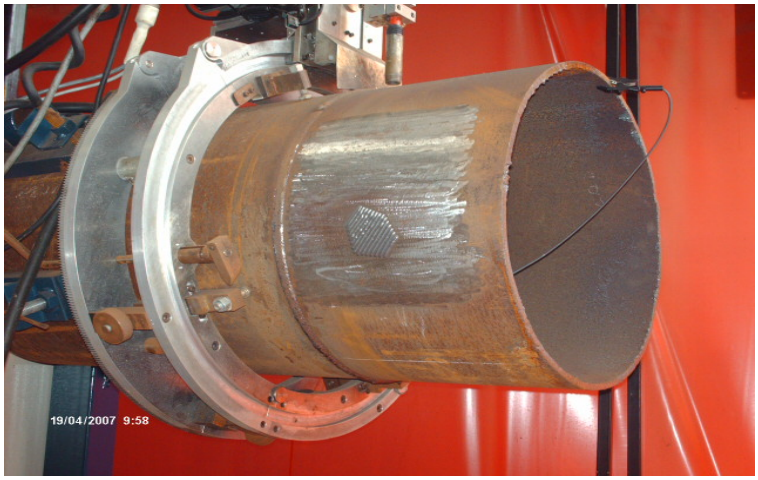


Figure 6.5-1. Pipe used as target for welding trials.

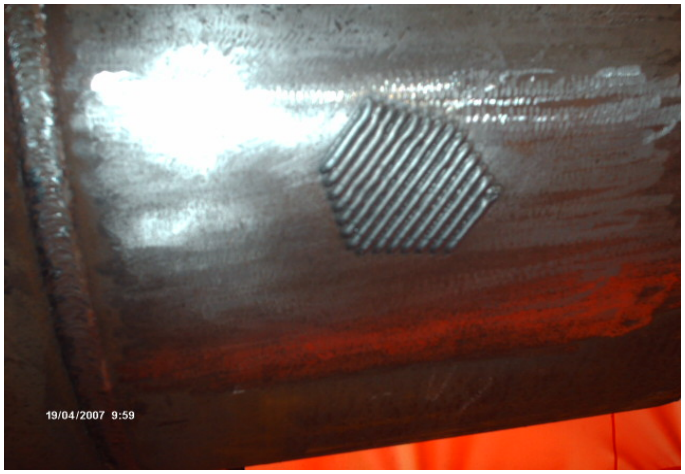


Figure 6.5-2. Weld trial 1 on pipe.

Figure 6.5-3. Weld trial 2 on pipe.

Figure 6.5-4. Weld trial 3 on pipe.

Figure 6.5-5. Various weld trials on pipe.

The weld trial in Figure 6.5-2 was carried out using a predefined inter-distance of 5mm between weld beads. In trials 3 and 4, Figures 6.5-3 and 6.5.4, the inter-distance was calculated automatically by the software according to the welding parameters defined in the power supply. Both were carried out with the same welding power and a bead overlap of 15%. However in trial 4 instability in the welding process can be seen (the welding lines do not look continuous). In Figure 6.5-5, a picture of the pipe with several different trials is shown. The welding deposit at the upper right was performed before welding trials two and three. In this trial, the inter-distance was too small creating a welding patch of excessive height and the heat generated during welding was too high due to the small distance between weld beads. In all of the weld trials on the pipe, the shape was correctly reproduced according to the robotic welding program.

To test the system on a more complex shape, a metallic plate was rolled to create a surface with two curves. The two curvatures on the plate were not symmetrical. This curved plate was placed on a grounded metallic work bench as shown in Figure 6.5.6.

Figure 6.5.7 shows two welding trials conducted on this work object. The large welding patch at the centre of the figure was done with a predefined weld bead inter-distance of 10 mm. The smaller weld patch at the bottom right was done with the inter-distance automatically calculated by the system with a no overlapping condition (0%). Both trials were satisfactory and the welding was performed correctly according to the area selected in the user interface, although the spacing of the beads in the first one would be unsuitable for

situations such as corrosion damage repair. More pictures of welding trials can be found in appendix B, additionally a set of pictures showing the touch sensing sequence can be found in AppendixD.

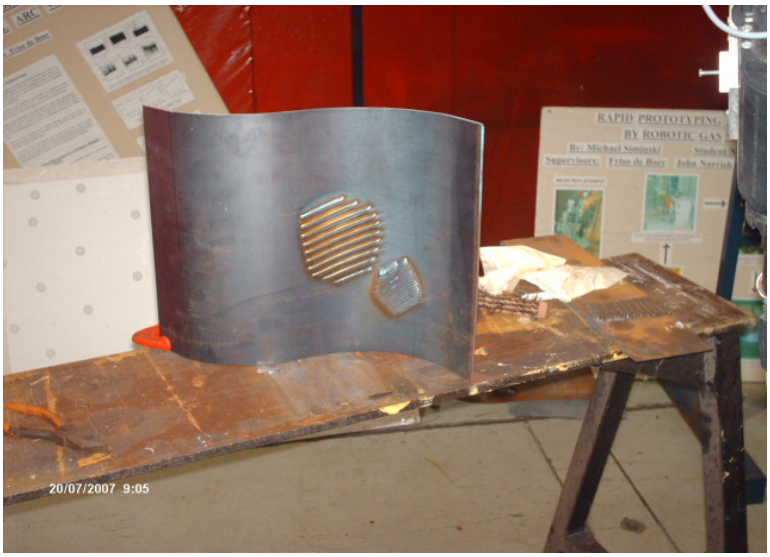


Figure 6.5-6. Work object with two curves.

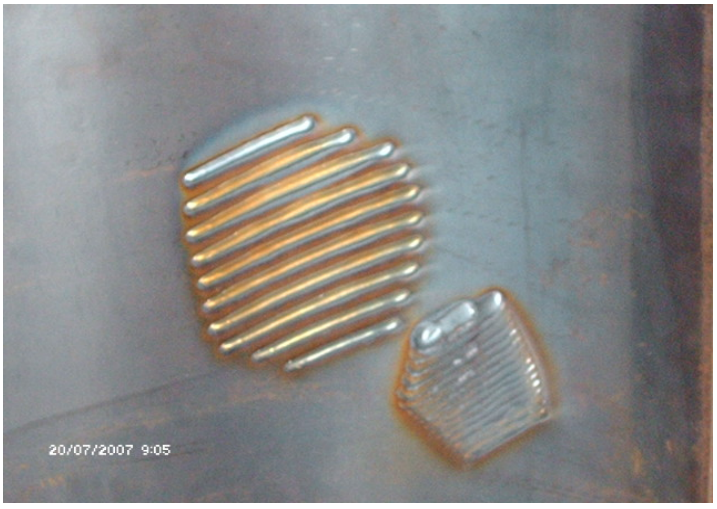


Figure 6.5-7. Weld trials on curved surface.

7. DISCUSSION

The current work was designed to investigate the feasibility of off line programming of robotic weld repair of curved surface profiles using a single camera vision system. The use of dual camera or laser profilometry techniques has previously been implemented for extracting surface topography but this approach is costly and has not been developed into an integrated system for rapidly generating robot programs for weld repair tasks. The system under investigation here was based on a single low cost camera and laser structured light was employed to generate artificial characteristic points (in order to extract the shape of the surface). An integrated software package was developed to convert the profile data and calculated deposition volumes into a robot program.

Whilst the x and y coordinates of the work area could be extracted from the system with sufficient accuracy for weld repair the z coordinate data initially contained large errors. Considerable emphasis was therefore attached to reducing potential profile generation errors and to calibration of the system. The calibration, analysis and correction of errors required the adoption of a novel and complex computation approach. Progressive testing was undertaken to evaluate the errors and the effect of the various correction strategies and a final validation trial was undertaken to prove the system during weld deposition on a simulated workpiece.

Discussion of the various aspects of the work has been provided throughout the preceding chapters, and in particular the evaluation of measurement errors which formed a basis for progressive development of the techniques has been described previously. However some additional comments are made below to clarify the methodology.

7.1 ARTIFICIAL FEATURE POINTS

In order to obtain the three dimensional shape of the surface of an object using a camera based vision system, it is necessary to extract characteristic features of the object. However, in many weld repair applications these characteristic features are not present or are not sufficiently distinct to obtain the desired information. To overcome this issue, artificial feature points generated by means of structured laser light were used. This choice was made based on the portability of laser projectors and their increased immunity to environmental conditions compared to other light sources. A structured laser module was built to produce a seven by seven dot matrix. Similar laser light patterns have been used by other authors for obtaining information of an object's surface such as the system proposed by Galván et al.[75] and the technique proposed by Saeed et al. [52]. Song and Zhang [76] also utilise a laser dot matrix pattern and propose some algorithms to process the image produced by its reflection from a weld pool. Initial attempts to use a matrix of individual low power laser diodes were abandoned due to the difficulty of ensuring accurate optical alignment. A single high power (90 mw) laser and an optical pattern generator were finally chosen and this proved to be very effective. The tests performed indicated that

with this laser diode the optical power output was sufficient for the current application.

Although laser light is monochromatic and is less affected by changes in environmental lighting, some problems arose under certain conditions, especially when the light source was reflected by the work object. This situation is illustrated in Figure 7.1-1 where the work object is a reflective metallic plate and a high intensity light source is aimed towards the work piece. The situation can be improved by decreasing the exposure time of the camera (shutter value) as shown in Figure 7.1-2. This does however reduce the number of feature points which may be extracted.

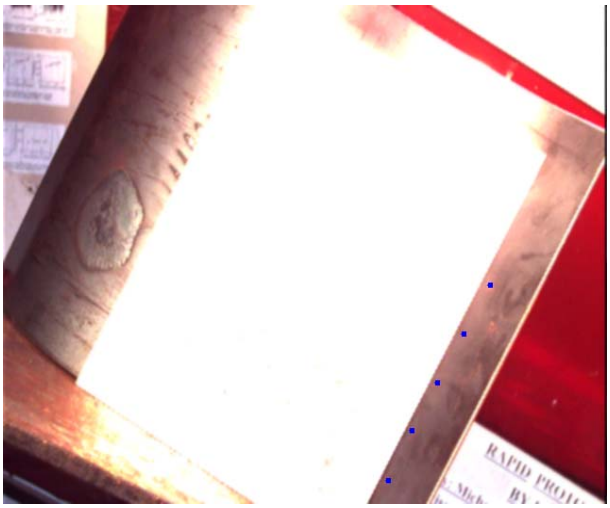


Figure 7.1-1. High intensity light source being reflected by work piece.

Another problem found with highly reflective surfaces is that they can generate high intensity specular reflections from the laser module itself. This situation is illustrated in Figure 7.1-3. Here some of the laser dots do notappear small and sharp but rather present an indistinct profile. It was found that some of the

reflections cause so much noise that it makes them undetectable to the software.

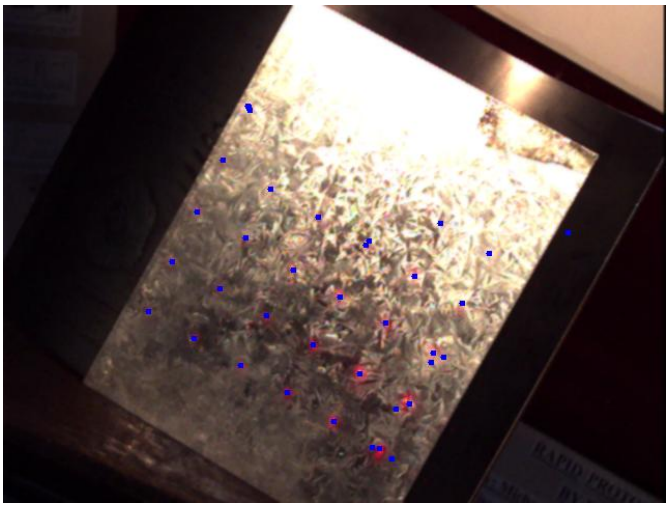


Figure 7.1-2. High intensity light source being reflected by work piece (shutter value reduced).

To avoid such undesirable reflections, the laser has to be placed at an angle with respect to the camera such that these reflections are not visible by the camera or at least do not overlap the feature points. An alternative illumination technique used to avoid specular reflection is to use polarising filters, one in front of the light source and another, with opposite polarization, in front of the camera [54]. The problem with this method is that it presents a high attenuation factor, i.e. typically less than 50% of the incident light passes through the polariser.

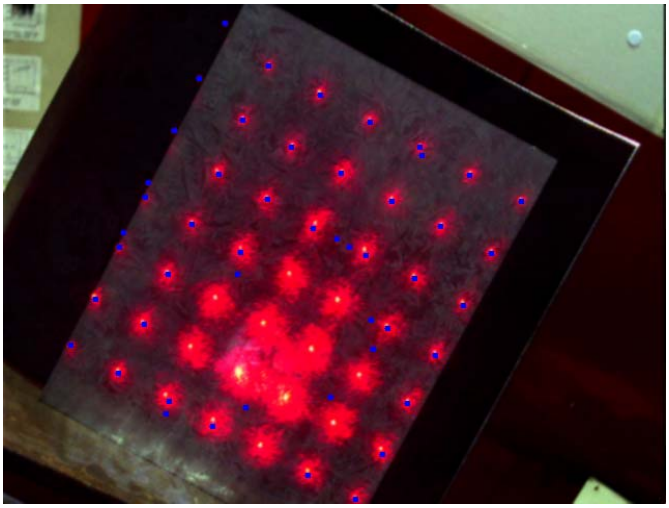


Figure 7.1-3. Laser module being reflected by working object.

7.2 CAMERA

The camera used for the experiment was a relatively high specification device (see Chapter 3) but was chosen because of its availability. However, the commercial cost of the camera was around AUD$1800. It should be stressed that it would be possible to use a lower cost camera that provides the functionality needed for the application but consideration would need to be given to optics as discussed above.

The basic requirements are: a digital camera with serial communications so that the image information can be retrieved by the computer. It is also required to be able to control different camera parameters such as shutter speed and brightness (not generally available on many low cost cameras). However, digital inputs and outputs, such as external triggers, are not required for this application. The frame rate of the camera is not a critical factor either. If the camera uses IEEE1394 connection, the camera should use the IIDC/DCAM

protocol so that a standard application can establish communication with the camera. Although a camera using a proprietary protocol may be able to communicate with the computer using proprietary applications and development libraries, the use of third party development tools like LabVIEW may be more difficult or not possible. Examples of alternative cameras are the *Guppy F-036* supplied by Allied Vision Technologies [77]. This camera has all the requirements mentioned above with an imaging device of 1/3'', it is smaller than the *Marlin* but costs less than half the price. The Imaging Source [78] also offers the *DFK 31F04* camera which has an IEEE1394 connection and imaging device of 1/4" for less than AUD$420.

7.2.1 Camera Calibration

Two calibration methods, the planes and the geometric (Tsai method) approaches have been evaluated in this work. Additionally a modification to the planes method was also developed to improve accuracy. The camera calibration does not introduce any additional processing stage to the robot program generation sequence since it is only required to calibrate the camera when it is first mounted on the robot arm or every time it is detached. The software developed in this thesis incorporates an optimised calibration routine.

The camera calibration can be affected by the accuracy of the calibration pattern and the accuracy with which the characteristic points are detected. When a paper calibration pattern was used, the accuracy of the results degraded with time whilst the paper changed size slightly due to the change in

humidity and temperature. For these reasons an accurately machined aluminium calibration grid was finally used. Changes in accuracy were also noticed depending on the environmental illumination, it was noticed that different illumination levels (without compensating for them by adjusting the camera parameters) could produce a change in the detection of the calibration points and therefore in the calibration parameters which are reflected in the accuracy of the vision system. Similar issues were reported by Tapper et al. [79] when evaluating the Tsai method. A detailed evaluation of how different aspects affect the calibration accuracy in the Tsai method has also been discussed by Sun and Cooperstock [80].

The modified planes method proposed uses the error factors to compensate for any possible source of error present in the normal planes calibration technique. Therefore it is not required to recalculate the value of f_x and f_y as it is with the standard planes calibration technique.

7.3 SURFACE MAPPING

A polynomial approach was chosen in this work to characterise the surface of the work piece. Although one of the most common methods used in different imaging applications is splines, this method was not used since the depth coordinate obtained from the vision technique has a significant amount of noise and is therefore not suitable for such a technique. It was considered that a third degree polynomial would be appropriate to characterise a diverse range of surfaces. Additionally, it was the aim of this work to concentrate on an area

manually defined by a user through the user interface. With this in mind, the equation obtained is intended to approximate the region of interest and not the whole surface of the object. However the method proposed allows definition of several repair jobs in the area covered by the laser dots (one after another) and requires only the execution of the vision extraction once and a touch sensing refinement for each job. In this way, repair jobs can be performed on objects with complex curved surfaces in contrast to other techniques like the one proposed by Madsen et al. [81] which requires a model of the object to be welded.

As previously shown in the 3D coordinate estimation results, the error obtained in the *z* axis is large enough to justify the use of the additional contact technique employed in this work. Using tactile sensing, an accurate *z* coordinate of selected points located in the area selected by the user is found and used to produce a new equation that satisfactorily describes the area of interest. In Figure 7.3-1, a test surface with the artificial characteristic dots is shown. To the right the 3D plot of the estimated coordinates is presented. 3D plots of the dot coordinates after filtering and the surface's equation has been calculated and are presented in Figure 7.3-2; the plot to the left is before tactile refinement and the one to right is after refinement.

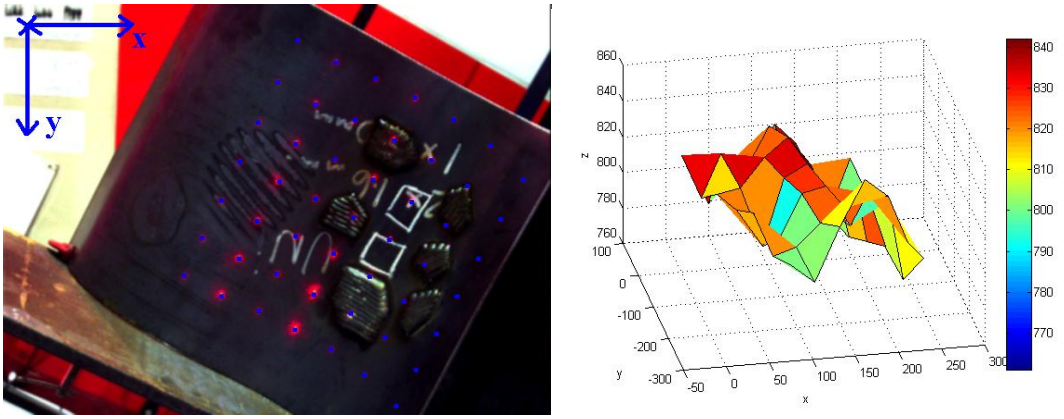


Figure 7.3-1. Extracted feature points (left) and plot of estimated 3D coordinates (mm) before filtering (right).

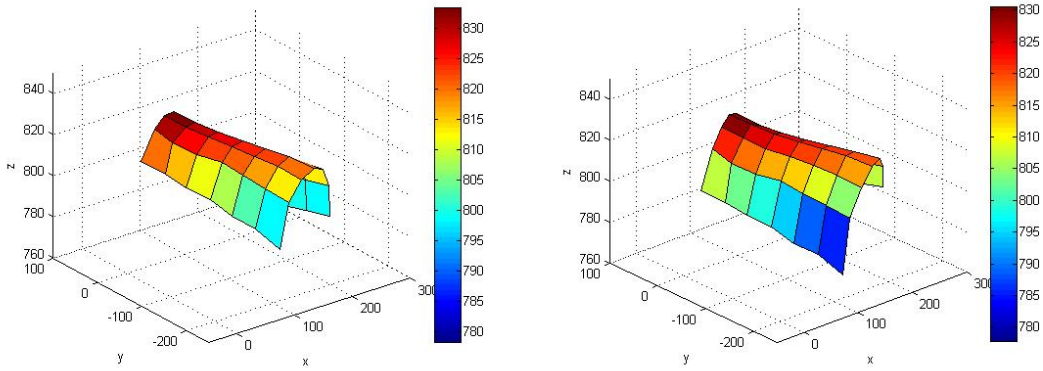


Figure 7.3-2. Plot of estimated 3D coordinates (mm) after filtering. Before tactile correction (left) and after correction (right).

7.4 WELD PATH GENERATION

, The current work uses a similar methodology to that reported by Nicholson[7] to calculate the weld paths. However, an additional stage had to be included in order to cope with curved surfaces; given that the straight lines which appear on the screen are really curved lines on the work piece surface and therefore their real length needs to be estimated in order to generate the weld runs of appropriate length. This stage corresponds to an “unfolding” or “developing” of

the surface to produce a two dimensional plot of the area of interest which includes the changes in the three dimensions.

Additionally, the orientation of the torch had to be estimated in order to produce acceptable welding quality. Adjustment of rotation around the torch *z* axis by the user was allowed. This feature allows the user to specify a more convenient orientation of the torch when a collision of the welding gun with the work piece or its surroundings is possible.

7.5 ROBOT PROGRAM CREATION

The generation of the robot program was undertaken with ABB's RAPID language syntax. A ".prg" file is created by the system and downloaded to the robot controller. After the tactile refinement sequence is performed, it takes only a few seconds to create the robot program file. The software application developed allows broad control of the different components of the system from a single application resulting in an easy to use and flexible interface. The communication with the power supply and the robot are established using the server applications and development libraries provided by the manufacturers. In this way the system's software is a client of such servers. The results presented in Chapter 6 show that the approach chosen to generate the curved welding lines in the robot program is adequate (by breaking them into smaller line segments) and produces satisfactory results although the generated robot programs are relatively large. Nevertheless, there is a big saving in time by using the proposed system; a manual program generation was tested for weld

repair of a small area (shown in Figure 7.5.1) and the total programming time was around 1 hour. Even with this small area the demands on the manual programmer are quite severe.

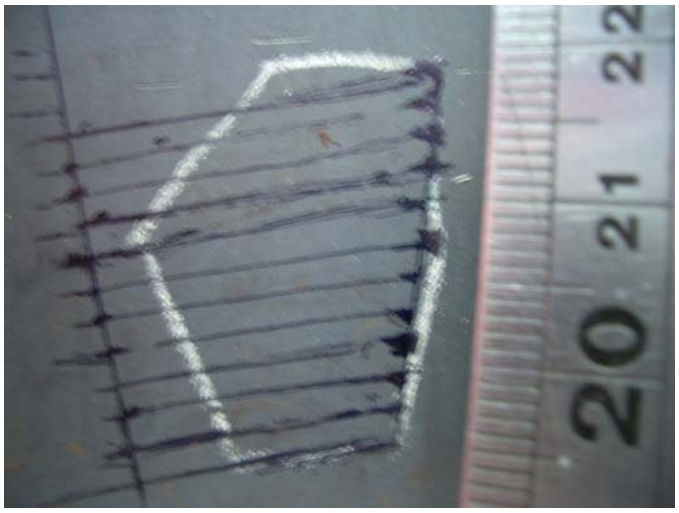


Figure 7.5-1. Area with lines drawn on it for manual program generation (point-to-point).

8. CONCLUSION AND RECOMMENDATIONS

8.1 CONCLUSIONS

1) A system for rapidly generating robot programs for weld repair tasks on complex surfaces has been described.
2) It has been demonstrated that it is possible to use a low cost single camera vision system and laser structured light to achieve acceptable accuracies for weld repair using gas metal arc welding.
3) Errors in determining the depth (z coordinate) were found to be the most significant issue and these were corrected using a combination of calibration, filtering, error checking and simple tactile sensing.
4) This combination produced faster surface contour acquisition than would be obtained by tactile sensing alone.
5) An integrated software package was developed to carry out surface profile measurement, weld deposition volume estimation and robot program generation.
6) Validation of the system was conducted by carrying out weld tests on curved surfaces and very acceptable results were obtained.
7) The proposed system is not tied to the knowledge of the geometry of the work piece, in contrast to other proposed systems for welding automation which employ a CAD model of the work-piece ; for example the system proposed by Young Chil [82] developed for ship building. The NOMAD project [83], which is a complex system composed of several sensors and a CAD model of the work-piece.

8.2 RECOMMENDATIONS

Real time control of torch to work piece distance would be a simple enhancement to this system which would further reduce errors In the case of welding repair, it has previously been demonstrated that this may be achieved by monitoring of the welding current and adaptively controlling the contact tip to work piece distance.

The design of the system makes it suitable for incorporation in a mobile robot or fully automatic mobile systems. Therefore, the study of a complete mobile system utilising the proposed system for weld repair on site is worth further consideration.

9. REFERENCES

[1] Czarnecki, C. A., "Design and off-line programming of a dual robot workcell for garment manufacture," *Microprocessors and Microsystems*, vol. 23, pp. 225, 1999.

[2] Dai, W. and Kampker, M., "PIN-a PC-based robot simulation and offline programming system using macro programming techniques," presented at The 25th Annual Conference of the IEEE Industrial Electronics Society, 1999. IECON '99 Proceedings. , 1999.

[3] Westkamper, E., Schraft, R. D., Schweizer, M., Herkommer, T. F., and Meiner, A., "Task-oriented programming of large redundant robot motion," *Robotics and Computer-Integrated Manufacturing*, vol. 14, pp. 363, 1998.

[4] Zuhlke, D., Mobius, F., and Schroder, C., "Symbols facilitate programming of industrial robots," presented at Proceedings., 1997 IEEE International Conference on Robotics and Automation, 1997. , Albuquerque, New Mexico, 1997.

[5] Ahn, C. K. and Lee, M. C., "An off-line automatic teaching by vision information for robotic assembly task," presented at IECON 2000. 26th Annual Confjerence of the IEEE Industrial Electronics Society, 2000. , 2000.

[6] Pan, J., "Vision Systems and Automatic path programming," in *Arc welding Control*. Cambridge: Woodhead Publishing and CRC Press, 2003, pp. 531-581.

[7] Nicholson, A., "Rapid Adaptive Programming Using Image Data", PhD thesis, Faculty of Engineering, University of Wollongong, Wollongong, Australia, 2005.

[8] Horn, D., "Machine Vision: The Guiding Light," *Mechanical Engineering*, vol. 111, pp. 40, 1989.

[9] Gregory, E. N., "Repair by Welding," in *Repair and Reclamation*. London: Welding Institute, 1986.

[10] Grainger, S., "Repair and reclamation surfacing by welding," in *Repair and Reclamation*. London: Welding Institute, 1986.

[11] Cary, H. B., *Modern welding technology / by Howard B. Cary*. Englewood Cliffs ; London etc. :: Prentice-Hall, 1979.

[12] Tung, P.-C., Wu, M.-C., and Hwang, Y.-R., "An image-guided mobile robotic welding system for SMAW repair processes," *International Journal of Machine Tools and Manufacture*, vol. 44, pp. 1223, 2004.

[13] Ltd, T., *Intelligent robot gains approval for ship repair [Online]*,Available:< http://www.twi.co.uk/j32k/protected/band_13/ship_case4.html >, [Accessed 2005].

[14] Sun, H., Wu, L., and Gao, H., "Remote welding robot system," presented at Robot Motion and Control, 2004. RoMoCo'04. Proceedings of the Fourth International Workshop on, 2004.

[15] Sprovieri, J., "Robots to Change the Future-Again," *Assembly*, vol. 50, pp. 52, 2007.

[16] Motoman Inc.,Available:< http://www.motoman.com/ >, [Accessed 2007].
[17] Inc., M., *MOTOMAN NEWS RELEASE [Online]*,Available:< http://www.motoman.com/motomedia/pr/DIA10.pdf >, [Accessed 2007].
[18] Angelo, J. A., *Robotics : a reference guide to the new technology / Joseph A. Angelo, Jr*. Westport, Conn. :: Greenwood Press, 2007.
[19] Spong, M. W., *Robot modeling and control / Mark W. Spong, Seth Hutchinson, M. Vidyasagar*. Hoboken, NJ :: John Wiley & Sons, 2006.
[20] Craig, J., *Introduction to Robotics*, Second ed: Addison Wesley, 1989.
[21] Fuller, J. L., *Robotics : introduction, programming, and projects / James L. Fuller*. New York :: Merrill, 1991.
[22] Carvalho, G. C., Siqueira, M. L., and Absi-Alfaro, S. C., "Off-line programming of flexible welding manufacturing cells," *Journal of Materials Processing Technology*, vol. 78, pp. 24, 1998.
[23] Joe, P., "Agile assembly with robots," *Manufacturing Engineering*, vol. 130, pp. 83, 2003.
[24] Kampker, M., "New ways of user-oriented robot programming," presented at Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, 1998. IECON '98., Aachen, 1998.
[25] Yoshimi, B. H. and Allen, P. K., "Alignment using an uncalibrated camera system," *IEEE Transactions on Robotics and Automation*, vol. 11, pp. 516, 1995.
[26] Wells, G., Venaille, C., and Torras, C., "Vision-based robot positioning using neural networks," *Image and Vision Computing*, vol. 14, pp. 715, 1996.
[27] Dhanesh, R. and Mandava, R., "Neural network-based robot visual positioning for intelligent assembly," *Journal of Intelligent Manufacturing*, vol. 15, pp. 219, 2004.
[28] Guerrero, J. J. and Sagues, C., "Uncalibrated vision based on lines for robot navigation," *Mechatronics*, vol. 11, pp. 759, 2001.
[29] Fofi, D., Salvi, J., and Mouaddib, E. M., "Uncalibrated reconstruction: An adaptation to structured light vision," *Pattern Recognition*, vol. 36, pp. 1631-1644, 2003.
[30] Gu, J. W. and Han, J. H., "3D reconstruction in a constrained camera system," *Pattern Recognition Letters*, vol. 23, pp. 1337-1347, 2001.
[31] Motta, J. M. S. T., de Carvalho, G. C., and McMaster, R. S., "Robot calibration using a 3D vision-based measurement system with a single camera," *Robotics and Computer-Integrated Manufacturing*, vol. 17, pp. 487, 2001.
[32] Tsai, R., "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* vol. 3, pp. 323-324, 1987.
[33] Sheng-Wen, S., Yi-Ping, H., and Wei-Song, L., "When should we consider lens distortion in camera calibration," *Pattern Recognition*, vol. 28, pp. 447, 1995.
[34] Weng, J., Cohen, P., and Herniou, M., "Camera calibration with distortion models and accuracy evaluation," *SO - IEEE Transactions on Pattern Analysis & Machine Intelligence* vol. v 14, pp. 965-980, 1992.
[35] Ma, L., Chen, Y., and Moore, K. L., *A Family of Simplified Geometric Distortion Models for Camera Calibration [Online]*,Available:< http://arxiv.org/abs/cs/0308003 >, [Accessed 2005].

[36] Gremban, K. D., Thorpe, C. E., and Kanade, T., "Geometric camera calibration using systems of linear equations," 1988.
[37] Champleboux, G., Lavallee, S., Sautot, P., and Cinquin, P., "Accurate calibration of cameras and range imaging sensor: the NPBS method," presented at Proceedings., 1992 IEEE International Conference on Robotics and Automation, 1992.
[38] Martins, H. A., Biirk, J. R., and Kelley, R. B., "Camera models Based on Data from Two Calibration Planes," *Computer Graphics and Image Processing*, pp. 173-180, 1981.
[39] Zhang, Z., "Flexible camera calibration by viewing a plane from unknown orientations," presented at Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, 1999.
[40] Park, S.-W. and Hong, K.-S., "Practical ways to calculate camera lens distortion for real-time camera calibration," *Pattern Recognition*, vol. 34, pp. 1199, 2001.
[41] Hecht, E., *Optics*, 4th ed ed. Reading, Mass. :: Addison-Wesley, 2002.
[42] Pedrotti, F. and Pedrotti, L., *Introduction to optics*, 2 ed. New Jersey: Prentice Hall, 1993.
[43] Zhang, Y. and Wang, Z., "A Flexible Camera Calibration Method for Computer Visual 3D Reconstruction System," presented at Signal Processing, The 8th International Conference on, 2006.
[44] Plataniotis, K. N. and Venetsanopoulos, A. N., *Color Image Processing and Applications*: Springer, 2000.
[45] National Instruments, "NI vision Concepts Manual," N. i. Corporation, Ed., 2005.
[46] Burdick, H. E., *Digital imaging : theory and applications / Howard E. Burdick*. New York :9701: McGraw-Hill, 1997.
[47] Netravali, A. N., *Digital pictures : representation, compression, and standards / Arun N. Netravali and Barry G. Haskell*, 2nd ed ed. New York :: Plenum Press, 1995.
[48] Wright, C., *YUV Colorspace [Online]*,Available:< http://softpixel.com/~cwright/programming/colorspace/yuv/ >, [Accessed 2007].
[49] Lee, M. H., Son, K., Han, S. H., Lee, J. M., Lee, M. C., Choi, J. W., Joo, H. N., and Chang, Y. H., "Development of an automated system for soldering and the solder inspection of an assembly line," presented at Proceedings of the 2000 IEEE International Symposium on Industrial Electronics, 2000.
[50] Ramachandram, D. and Rajeswari, M., "Structured lighting to enhance global image feature sensitivity in a neural network based robot-positioning task," presented at TENCON 2000. Proceedings Kuala Lumpur 2000
[51] Wong, A. K. C., Niu, P., and He, X., "Fast acquisition of dense depth data by a new structured light scheme," *Computer Vision & Image Understanding*, vol. 98, pp. 398-422, 2005.
[52] Saeed, G., Lou, M., and Zhang, Y. M., "Computation of 3D weld pool surface from the slope field and point tracking of laser beams," *Measurement Science and Technology*, pp. 389, 2003.

[53] Skydan, O. A., Lalor, M. J., and Burton, D. R., "Using coloured structured light in 3-D surface measurement," *Optics & Lasers in Engineering*, vol. 44, pp. 65-78, 2005.
[54] Melles Griot, *Machine Vision Guide [Online]*,Available:< http://machinevision.mellesgriot.com/mvguide/lif_3.htm >, [Accessed 2006].
[55] IVP, S., *Ranger - 3D Camera [Online]*,Available:< http://www.sick.com/home/factory/news/industrial/visionsensor/ranger/en.html >.
[56] SICK IVP,Available:< http://www.sickivp.se/sickivp/en.html >, [Accessed 2006].
[57] Renishaw plc,Available:< http://www.renishaw.com/en/1030.aspx >, [Accessed 2007].
[58] Reishaw, *OTP6M non-contact probe [Online]*,Available:< http://www.renishaw.com/en/6666.aspx >, [Accessed january 2007].
[59] DGA, R., *Scanners [Online]*,Available:< http://www.rolanddga.com/asd/products/scanners/LPX600/default.asp?menu=FEATURES&submenu=7&nxt=2&totsubcnt=7 >, [Accessed 2007].
[60] FARO Technologies Inc.,Available:< http://www.faro.com/ >, [Accessed 2007].
[61] ROMER,Available:< http://www.romer.com/ >, [Accessed 2007].
[62] Hexagon Metrology, *ROMER [Online]*,Available:< http://fr.romer.com/ >, [Accessed 2007].
[63] FARO Technologies Inc., *FaroArm [Online]*,Available:< http://www.faro.com/content.aspx?ct=us&content=pro&item=2 >, [Accessed 2007].
[64] SICK, *Prximity Lase Scanner PLC - Technical Description [Online]*,Available:< http://www.mysick.com/saqqara/pdf.aspx?id=im0011885 >, [Accessed 2007].
[65] Technologies, F., *P-scan System 4 [Online]*,Available:< http://www0.force.dk/p-scan/ >, [Accessed 2007].
[66] The Imaging Source, *Lenses: Selection and Setup [Online]*,Available:< http://www.theimagingsource.com/en/resources/whitepapers/download/choosinglenswp.en.pdf >, [Accessed 2007].
[67] Allied Vision Technologies, *AVT Marlin: Technical Manual [Online]*,Available:< http://www.alliedvisiontec.com/fileadmin/files/pdf/produkte/technical_manuals/MARLIN_TechMan_V2.2.0_en.pdf >, [Accessed 2005].
[68] Fronius International, *The new revolution in digital GMA welding [Online]*,Available:< http://www.fronius.com/cps/rde/xchg/SID-0AFF0106-B5425DC6/fronius_international/hs.xsl/79_1997_ENG_HTML.htm >, [Accessed 2006].
[69] Sid-Ahmed, M. A., *Image Processing: Theory, algorithms and architectures*: McGraw-Hill, Inc., 1995.
[70] National Instruments, *LabVIEW [Online]*,Available:< http://www.ni.com/labview/ >, [Accessed 2006].
[71] The MathWorks, *MATLAB [Online]*,Available:< http://www.mathworks.com/products/matlab/ >, [Accessed 2006].

[72] Microsoft Corporation, *Visual Basic [Online]*,Available:< http://msdn.microsoft.com/es-co/vbasic/default.aspx >, [Accessed 2006].
[73] Microsoft Corporation, *Visual Studio [Online]*,Available:< http://msdn.microsoft.com/es-co/vstudio/default.aspx >, [Accessed 2007].
[74] Stockeryale, *Laser FAQs [Online]*,Available:< http://www.stockeryale.com/i/lasers/faqs.htm#1mw >, [Accessed 2006].
[75] Gonzalez-Galvan, E. J., Cruz-Ramirez, S. R., Seelinger, M. J., and Cervantes-Sanchez, J. J., "An efficient multi-camera, multi-target scheme for the three-dimensional control of robots using uncalibrated vision," *Robotics and Computer-Integrated Manufacturing*, vol. 19, pp. 387, 2003.
[76] Song, H. S. and Zhang, Y. M., "Image Processing for Measurement of Three-Dimensional GTA Weld Pool Surface," *Welding Journal*, vol. 86, pp. 323-330, 2007.
[77] Allied Vision Technologies GmbH,Available:< http://www.alliedvisiontec.com/ >, [Accessed 2005].
[78] The Imaging Source Europe GmbH,Available:< http://www.theimagingsource.com/en/products/ >, [Accessed 2006].
[79] Tapper, M., McKerrow, P., and Abrantes, J., "Problems encountered in the implementation in the Tsai's algorithm for camera calibration," presented at Australasian conference on robotics and Automation, 2002.
[80] Sun, W. and Cooperstock, J. R., "Requirements for Camera Calibration: Must Accuracy Come with a High Price?," presented at Application of Computer Vision, 2005. WACV/MOTIONS '05 Volume 1. Seventh IEEE Workshops on, 2005.
[81] Ole, M., Carsten Bro, S., Rune, L., Lars, O., and Niels, J. J., "A system for complex robotic welding," *The Industrial Robot*, vol. 29, pp. 127, 2002.
[82] Young Chil, D., Jae Chang, L., Seung Hwan, M., Gi Beom, P., Mi Hee, N., Hyun Cheol, L., Do Hyung, J., Dae Kyung, K., Se Hwan, K., Jae Woong, Y., and Tae Ho, K., "Camera Vision System for Automation of Sub-assembly Line in Shipbuilding," presented at SICE-ICASE, 2006. International Joint Conference, 2006.
[83] *Robotic arc welding - successful conclusion to European project [Online]*,Available:< <http://www.twi.co.uk/j32k/unprotected/band_1/c1342.html> >, [Accessed 2005].

10. BIBLIOGRAPHY

[1] *Quaternions and spatial rotation [Online]*,Available:< http://www.sciencedaily.com/encyclopedia/quaternions_and_spatial_rotation >.

[2] *Three-Dimensional transformations [Online]*,Available:< http://cs.anu.edu.au/escience/lecture/cg/Transformation/printNotes.en.html >, [Accessed 2005].

[3] *Development and validation of reliability forecast techniques for selected thermomechanical components of gas turbine combined-cycle plants : guidelines for making reliability forecasts : topical report, May 1981 / prepared by Westinghouse Electric Corporation ; principal investigators R.E. Strong, K.H. Eagle ; prepared for Electric Power Research Institute*. Palo Alto, Calif :: The Institute, 1981.

[4] *Repair of steel pipelines / [editor: Ben Gross]*, 1st ed ed. Lidcombe, N.S.W. :: Welding Technology Institute of Australia, 1994.

[5] "Giving robots eyes and hands [Visual servoing]," *Plant*, vol. 59, pp. 13, 2000.

[6] *Intelligent robot gains approval for ship repair [Online]*,Available:< http://www.twi.co.uk/j32k/protected/band_13/ship_case4.html >, [Accessed 2005].

[7] *How color cameras work - Principles [Online]*,Available:< http://www.1394imaging.com/resources/backgnd/cameras/color/camera_basics/?sid=a10884b3b7866beac549593efda91975 >, [Accessed 2005].

[8] ABB, "IRB 1410 Industrial Robot," ABB Automation Technologies, 2006.

[9] Adam, P., "A match made in robot heaven," *Plant*, pp. 18, 2004.

[10] Andrew, W., "Smart sensors vie for vision applications," *Vision Systems Design*, vol. 11, pp. 35, 2006.

[11] Anonymous, "Welding of body scanner vessels improved using robots vision system," *Assembly Automation*, vol. 23, pp. 198, 2003.

[12] Armel, C. and Francois, C., "Visual servoing based on image motion," *International Journal of Robotics Research*, vol. 20, pp. 857, 2001.

[13] Bang, B. I.-W., Son, Y.-P., Oh, K. H., Kim, Y.-P., and Kim, W.-S., "Numerical simulation of sleeve repair welding of In-service gas pipelines," *Welding Journal*, vol. December, pp. 273-282, 2002.

[14] Bauchspiess, A., Absi Alfaro, S. C., and Dobrzanski, L. A., "Predictive sensor guided robotic manipulators in automated welding cells," *Journal of Materials Processing Technology*, vol. 109, pp. 13, 2001.

[15] Bishop, B. E. and Spong, M. W., "Adaptive calibration and control of 2D monocular visual servo systems," *Control Engineering Practice*, vol. 7, pp. 423, 1999.

[16] Bradley, J. C., *Programming in Visual Basic version 6.0 update edition / Julia Case Bradley, Anita C. Millspaugh*. Boston ; London :: McGraw-Hill Irwin, 2002.

[17] Cardillo, J. and Sid-Ahmed, M. A., "A 3-D robot vision system using passive focus information," *Computers in Industry*, vol. 15, pp. 317, 1990.

[18] Chen, C.-S., Hung, Y.-P., Chiang, C.-C., and Wu, J.-L., "Range data acquisition using color structured lighting and stereo vision," *Image & Vision Computing*, vol. 15, pp. 445-456, 1997.

[19] Chen, J.-M., "A genetic-based vision system for cross-functional integration in flexible manufacturing: A tutorial and application," *International Journal of Flexible Manufacturing Systems*, vol. 9, pp. 343, 1997.

[20] Chen, S. Y., Li, Y. F., and Jianwei, Z., "Fast 3D Perception by a Color-Coded Vision System," presented at Mechatronics and Automation, Proceedings of the 2006 IEEE International Conference on, 2006.

[21] Chen, Z., Tseng, D.-C., and Lin, J.-Y., "A simple vision algorithm for 3-D position determination using a single calibration object," *Pattern Recognition*, vol. 22, pp. 173, 1989.

[22] Cooperative Researh Centre for Welded Structures, *In-service welding of high pressure gas pipelines [Online]*,Available:< <http://www.crcws.com.au/research/pipelines.htm> >, [Accessed 2005].

[23] Dai, W. and Kampker, M., "User oriented integration of sensor operations in a offline programming system for welding robots," presented at Proceedings. ICRA '00. IEEE International Conference on Robotics and Automation, 2000. , 2000.

[24] Demeyere, M., Dereine, E., Eugene, C., and Naydenov, V., "Measurement of cylindrical objects through laser telemetry: Application to a new forest caliper," *IEEE Transactions on Instrumentation & Measurement*, vol. 51, pp. 645-649, 2002.

[25] Edison Welding Institute, I., *Welding In-Service Pipelines [Online]*,Available:< http://www.ewi.org/technologies/arcwelding/pipelines.asp >, [Accessed 2005].

[26] Fabio, Benedetto, and Carlo, "Hybrid visual servoing: A combination of nonlinear control and linear vision," *Robotics and Autonomous Systems*, vol. 29, pp. 243, 1999.

[27] Faro Technologies Inc., *Laser Scanner LS [Online]*,Available:< http://www.faro.com/content.aspx?ct=us&content=pro&item=5 >, [Accessed 2007].

[28] Fonius, "TransPuls Synergyc 3200/4000/5000 Operating Instructions," Fronius, 2004.

[29] Forsyth, D., *Computer vision : a modern approach / David A. Forsyth, Jean Ponce*. Upper Saddle River, N.J. ; London :: Prentice Hall, 2003.

[30] Freund, E. and Rokossa, D., "Automatic trajectory genertation for Multi-Joint Robot Systems," presented at Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998.

[31] Ganapathy, S., "Decomposition of transformation matrices for robot vision," *Pattern Recognition Letters*, vol. 2, pp. 401, 1984.

[32] Garg, D. P., "Object Classification via stereo vision sensing i a flexible manufacturing work cell," in *Mechatronics and Machine Vision 2003: Future trends*, J. Billingsley, Ed. Baldock, Haertfordshire, England: Research Studies Press LTD., 2003, pp. 195-209.

[33] Gonzalez-Galvan, E. J., Skaar, S. B., and Seelinger, M. J., "Efficient camera space target disposition in a matrix of moments structure using camera-space manipulation," *The international Journal of Robotics Research*, vol. 18, pp. 809-818, 1999.

[34] Hager, G. D., "A modular system for robust positioning using feedback from stereo vision," *IEEE Transactions on Robotics and Automation*, vol. 13, pp. 582, 1997.
[35] Hongwei, G., Chengdong, W., Lifu, G., and Bin, L., "An Improved Two-Stage Camera Calibration Method," presented at Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on, 2006.
[36] Horaud, R., Mohr, R., and Lorecki, B., "Linear camera calibration," presented at Proceedings of the1992 IEEE International Conference on Robotics and Automation, 1992, 1992.
[37] Hu, Y., Xi, J., Yang, Z., Li, E., and Chicharo, J., "Generalized Analysis Model for Fringe Pattern Profilometry," presented at Instrumentation and Measurement Technology Conference, 2005. IMTC 2005. Proceedings of the IEEE, 2005.
[38] Hui, Z., Kwan-Yee, K. W., and Guoqiang, Z., "Camera Calibration from Images of Spheres," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 499-502, 2007.
[39] Hutchinson, S., Hager, G. D., and Corke, P. I., "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 651, 1996.
[40] Hwang, C.-p. and Ho, C.-S., "A knowledge-based task-level programming and execution environment for robots," *Robotics and Computer-Integrated Manufacturing*, vol. 12, pp. 329, 1996.
[41] Iccvg, *Computer vision and graphics : International Conference, ICCVG 2004, Warsaw, Poland, September 2004, proceedings / edited by K. Wojciechowski*. Dordrecht, The Netherlands :: Springer, 2006.
[42] Isak, K., Samir, V., and Vlatko, D., "3D Vision in Industrial Robot Working Process," presented at 12th International Power Electronics and Motion Control Conference, 2006.
[43] Ito, M. and Ishii, A., "A non-iterative procedure for rapid and precise camera calibration," *Pattern Recognition*, vol. 27, pp. 301, 1994.
[44] Jean-Paul, B., "Robot vision helps ensure weld quality," *Manufacturing Engineering*, vol. 119, pp. 20, 1997.
[45] Ji, Z., Leu, M. C., and Lilienthal, I. P. F., "Vision based tool calibration and accuracy improvement for assembly robots," *Precision Engineering*, vol. 14, pp. 168, 1992.
[46] Jitae, K. and Suck-Joo, N., "Vision sensor-based measurement for automatic die remodeling," *Journal of Manufacturing Systems*, vol. 22, pp. 73, 2003.
[47] John, S., "Robots to Change the Future-Again," *Assembly*, vol. 50, pp. 52, 2007.
[48] Kim, M. Y., Cho, H. S., and Kim, J. H., "A 3D sensor system using multi-stripe laser and stereo camera for environment recognition of mobile robots," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 4902, pp. 542-552, 2002.
[49] King, R., Ahmadi, M., Gorgui, R., Kwabwe, A., and Azimi, M., *Digital filtering in one and two dimensions : design and applications / Robert King ... [et al.]*. New York :: Plenum Press, 1989.
[50] Kite, D. H. and Magee, M., "Determining the 3D position and orientation of a robot camera using 2D monocular vision," *Pattern Recognition*, vol. 23, pp. 819, 1990.

[51] Kurfess, T., *Robotics and Automation Habdbook*: CRC Press, 2005.
[52] Li, L., Chen, S., and Lin, T., "3D vision technology and its applications in welding," presented at Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th, 2004.
[53] Lu, T.-f. and Lin, G. C. I., "An on-line relative position and orientation error calibration methodology for workcell robot operations," *Robotics and Computer-Integrated Manufacturing*, vol. 13, pp. 89, 1997.
[54] Maddalena, D., Zampato, M., and Favaretto, M., "A stereovision system for underwater, remotely operated vehicles," *Sensor Review*, vol. 18, pp. 121, 1998.
[55] Mahajan, A. and Figueroa, F., "An automatic self-installation and calibration method for a 3D position sensing system using ultrasonics," *Robotics and Autonomous Systems*, vol. 28, pp. 281, 1999.
[56] Malis, E., Chaumette, F., and Boudet, S., "2-1/2-D visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, pp. 238, 1999.
[57] Michael, V., "Robots with vision," *Mechanical Engineering*, vol. 120, pp. 26, 1998.
[58] Michel, D., "Renault develops innovative arc welding solutions for exhaust systems that cut production costs by 30 per cent," *The Industrial Robot*, vol. 29, pp. 318, 2002.
[59] Mike, W., "Vision systems in the automotive industry," *The Industrial Robot*, vol. 26, pp. 354, 1999.
[60] Mohamed, R., Ahmed, A., Eid, A., and Farag, A., "Support Vector Machines for Camera Calibration Problem," presented at Image Processing, 2006 IEEE International Conference on, 2006.
[61] Mulligan, J., "Fast Calibrated Stereo Vision for Manipulation," *Real-Time Imaging*, vol. 3, pp. 331, 1997.
[62] Oppenheim, A. and Schafer, R., *Digital Signal Processing*. New Jersey: Prentice-Hall, 1975.
[63] Park, S.-W., Seo, Y., and Hong, K.-S., "Real-Time Camera Calibration for Virtual Studio," *Real-Time Imaging*, vol. 6, pp. 433, 2000.
[64] Pauli, J., Schmidt, A., and Sommer, G., "Vision-based integrated system for object inspection and handling," *Robotics and Autonomous Systems*, vol. 37, pp. 297, 2001.
[65] Press, W. H., *Numerical recipes in C++ : the art of scientific computing* 2nd ed: Cambridge University Press, 2002.
[66] Rawlings, J. O., *Applied regression analysis : a research tool / John O. Rawlings*. Pacific Grove, Calif. :: Wadsworth & Brooks/Cole Advanced Books & Software, 1988.
[67] Ricolfe-Viala, C. and Sanchez-Salmeron, A. J., "Improving accuracy and confidence interval of camera parameters estimated with a planar pattern," presented at Image Processing, 2005. ICIP 2005. IEEE International Conference on, 2005.
[68] Ryberg, A., Christiansson, A. K., Eriksson, K., and Lennartson, B., "A new Camera Model and Algorithms for higher Accuracy and better Convergence in Vision-based Pose Calculations," presented at Mechatronics and Automation, Proceedings of the 2006 IEEE International Conference on, 2006.

[69] Salvi, J., Batlle, J., and Mouaddib, E., "A robust-coded pattern projection for dynamic 3D scene measurement," *Pattern Recognition Letters*, vol. 19, pp. 1055, 1998.
[70] Shafi, A., "Robots and vision combine for application in plastics industries," *Industrial Robot: An International Journal* vol. 31, pp. 351-354, 2004
[71] Song, L., Qu, X., Yang, Y., Chen, Y., and Ye, S., "Application of structured lighting sensor for online measurement," *Optics & Lasers in Engineering*, vol. 43, pp. 1118-1126, 2005.
[72] Stephens, R., *Visual Basic graphics programming : hands-on applications and advanced color development / Rod Stephens*, 2nd ed ed. New York ; Chichester :: Wiley Computer Pub, 2000.
[73] Sun, H., Wu, L., and Gao, H., "Remote welding robot system," presented at Robot Motion and Control, 2004. RoMoCo'04. Proceedings of the Fourth International Workshop on, 2004.
[74] Takeno, J. and Rembold, U., "Stereovision systems for autonomous mobile robots," *Robotics and Autonomous Systems*, vol. 18, pp. 355, 1996.
[75] Tarabanis, K., Tsai, R. Y., and Goodman, D. S., "Calibration of a Computer Controlled Robotic Vision Sensor with a Zoom Lens," *CVGIP: Image Understanding*, vol. 59, pp. 226, 1994.
[76] Toth, E. and Tel, F., "Calibrated virtual reality supported by stereo vision in intelligent robot control system," presented at Proceedings of the IEEE International Symposium on Industrial Electronics, 1999, 1999.
[77] Wu, C. S., Gao, J. Q., Liu, X. F., and Zhao, Y. H., "Vision-based measurement of weld pool geometry in constant-current gas tungsten arc welding," *Proceedings of the Institution of Mechanical Engineers*, vol. 217, pp. 879, 2003.
[78] Wu, Q. M. J., Lee, M.-F. R., and De Silva, C. W., "An imaging system with structured lighting for on-line generic sensing of three-dimensional objects," *Sensor Review*, vol. 22, pp. 46-50, 2002.
[79] Xu, Q., Ye, D., Che, R., and Huang, Y., "Accurate Camera Calibration with New Minimizing Function," presented at Robotics and Biomimetics, 2006. ROBIO '06. IEEE International Conference on, 2006.
[80] Yongjie, Y., Qidan, Z., Zhuang, L., and Quanfu, C., "Camera Calibration in Binocular Stereo Vision of Moving Robot," presented at Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on, 2006.
[81] Yoo, W.-S. and Na, S.-J., "Determination of 3D weld seams in ship blocks using a laser vision sensor and a neural network," *Juournal of Manufacturing Systems*, vol. 22, pp. 340-347, 2003.
[82] Zhang, G., He, J., and Yang, X., "Calibrating camera radial distortion with cross-ratio invariability," *Optics & Laser Technology*, vol. 35, pp. 457, 2003.
[83] Zhang, Z., *A Flexible New Technique for Camera Calibration [Online]*, Technical Report MSRTR- 98-71, Microsoft Research,Available:< http://research.microsoft.com/~zhang/Calib/ >,[Accessed: March 2006].
[84] Zhuang, H., Roth, Z. S., Xu, X., and Wang, K., "Camera calibration issues in robot calibration with eye-on-hand configuration," *Robotics and Computer-Integrated Manufacturing*, vol. 10, pp. 401, 1993.

APPENDIX A.

The 3D coordinate estimation errors from two sets of data are included in the following tables:

Table A1. Data set 1 of errors (mm) obtained using different camera calibration methods and different extraction techniques

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method[(1)]	Error type[(2)]	Dist.
0.4774	0.5822	3.1597	0.8859	1.0742	7.5399	plan(f)	geo	350
0.690788	0.458981	3.669919	1.318438	1.040869	8.293779	plan(f)	vect-geo	350
0.362	0.5775	3.7286	0.8237	1.0033	8.0064	plan(f)	vect	350
0.2923	0.3104	2.7129	0.6004	0.6672	4.6823	geom	geo	350
0.2207	0.334	4.4169	0.5955	0.7631	5.9186	geom	vect-geo	350
0.2263	0.2714	3.2702	0.573	0.6639	5.0439	geom	vect	350
0.1837	0.5191	6.3263	0.4337	0.8797	8.2694	plan(squ)	geo	350
0.1806	0.5369	8.1861	0.4633	1.0034	9.6769	plan(squ)	vect-geo	350
0.1434	0.4779	6.9316	0.3938	0.9035	8.6804	plan(squ)	vect	350
0.4548	0.5384	3.1608	0.858	1.0111	7.5388	plan(fo)	geo	350
0.304	0.572	4.901	0.8488	1.0402	8.9378	plan(fo)	vect-geo	350
0.3423	0.5282	3.7291	0.8035	0.9521	8.0085	plan(fo)	vect	350
0.2588	0.747	8.4004	0.9547	1.5219	13.0571	plan(f)	geo	400
0.451756	0.629058	4.441224	0.913091	1.074957	9.578089	plan(f)	vect-geo	400
0.2658	0.814	7.465	0.9881	1.5385	12.0103	plan(f)	vect	400
0.3362	0.5106	7.8335	0.7996	1.4686	10.7124	geom	geo	400
0.2514	0.4309	5.2049	0.6628	1.2	8.0957	geom	vect-geo	400
0.2851	0.5072	6.9	0.6919	1.3853	9.7524	geom	vect	400
0.391	0.6953	11.842	0.934	1.7072	14.7018	plan(squ)	geo	400
0.2524	0.6528	9.3576	0.6702	1.4654	12.2321	plan(squ)	vect-geo	400
0.3087	0.7277	10.9621	0.7595	1.6504	13.7966	plan(squ)	vect	400
0.2575	0.7077	8.3994	0.9269	1.4825	13.0562	plan(fo)	geo	400
0.2585	0.6759	5.7705	0.8919	1.3355	10.2726	plan(fo)	vect-geo	400
0.2573	0.7663	7.4656	0.9679	1.4875	12.0134	plan(fo)	vect	400
0.5899	0.6079	13.0165	1.614	1.4618	18.6805	plan(f)	geo	450
0.2807	0.312445	4.299444	0.858612	0.816773	10.23676	plan(f)	vect-geo	450
0.3298	0.6143	10.2492	1.0716	1.3549	15.7568	plan(f)	vect	450
0.7135	0.5196	12.2949	1.8081	1.409	14.3702	geom	geo	450
0.2622	0.278	4.9865	0.9113	0.7053	7.6266	geom	vect-geo	450
0.4227	0.4263	9.5465	1.3157	1.2046	11.7218	geom	vect	450
0.8702	0.5899	16.6935	1.9729	1.6725	18.7434	plan(squ)	geo	450
0.3531	0.3663	9.5152	1.0978	0.9941	12.1526	plan(squ)	vect-geo	450
0.5377	0.557	14.0025	1.4892	1.4937	16.1698	plan(squ)	vect	450
0.6026	0.5865	13.0129	1.6306	1.4214	18.6736	plan(fo)	geo	450
0.2105	0.3732	5.6652	0.8329	0.8276	11.026	plan(fo)	vect-geo	450
0.3354	0.5789	10.25	1.0918	1.3039	15.7602	plan(fo)	vect	450
0.7256	0.5361	14.4491	1.7119	1.2598	21.5544	plan(f)	geo	500
0.203931	0.171375	3.632821	0.733652	0.661208	9.32864	plan(f)	vect-geo	500
0.3643	0.4545	10.8445	0.9889	1.012	17.0702	plan(f)	vect	500
0.8541	0.5444	13.5619	1.9306	1.4072	16.7279	geom	geo	500

0.2528	0.3276	4.2267	0.7587	0.8217	6.5822	geom	vect-geo	500
0.4671	0.3953	9.992	1.2524	0.9371	12.7781	geom	vect	500
1.0681	0.5389	18.3303	2.1336	1.3732	21.4944	plan(squ)	geo	500
0.4282	0.259	9.1263	0.9997	0.6217	11.4774	plan(squ)	vect-geo	500
0.6466	0.4335	14.8217	1.4637	1.173	17.6062	plan(squ)	vect	500
0.7414	0.5292	14.4445	1.728	1.3112	21.5453	plan(fo)	geo	500
0.1781	0.2446	5.0367	0.7042	0.7406	10.1913	plan(fo)	vect-geo	500
0.3749	0.4311	10.8456	1.0091	0.961	17.0741	plan(fo)	vect	500
0.7697	0.4394	12.5321	1.4985	1.437	18.6599	plan(f)	geo	550
0.137657	0.358077	2.403181	0.304569	0.849533	7.413821	plan(f)	vect-geo	550
0.3906	0.2879	8.9763	1.0207	1.0083	14.669	plan(f)	vect	550
0.909	0.6049	11.4787	1.7501	1.5967	14.3566	geom	geo	550
0.3064	0.6066	2.3047	0.9308	1.0045	4.8491	geom	vect-geo	550
0.5072	0.4489	7.9672	1.2883	1.1374	10.131	geom	vect	550
1.1926	0.4512	16.595	1.9962	1.2906	19.4612	plan(squ)	geo	550
0.6001	0.3119	7.4947	1.1977	0.715	10.0932	plan(squ)	vect-geo	550
0.7911	0.3062	13.1493	1.5459	0.8462	15.3016	plan(squ)	vect	550
0.7882	0.4519	12.5277	1.5179	1.4891	18.6526	plan(fo)	geo	550
0.2	0.3346	3.2551	0.6753	0.9282	8.3152	plan(fo)	vect-geo	550
0.4067	0.2924	8.9775	1.0408	1.0601	14.6732	plan(fo)	vect	550
0.7941	0.3957	11.6026	1.3685	1.3488	18.0678	plan(f)	geo	600
0.171242	0.339804	2.374185	0.621148	0.71199	6.063672	plan(f)	vect-geo	600
0.3873	0.2395	7.7262	0.6822	0.8735	13.4323	plan(f)	vect	600
0.9327	0.5726	10.3766	1.5344	1.5216	14.6879	geom	geo	600
0.3137	0.5975	1.1589	0.6945	0.9389	4.6055	geom	vect-geo	600
0.511	0.4163	6.557	0.9664	1.0154	10.7664	geom	vect	600
1.2549	0.3998	15.8468	1.8329	1.1916	20.1493	plan(squ)	geo	600
0.6456	0.2717	5.9681	1.0324	0.6216	10.222	plan(squ)	vect-geo	600
0.8424	0.2538	12.0973	1.3215	0.7652	16.2987	plan(squ)	vect	600
0.8136	0.4079	11.5978	1.3797	1.4017	18.0583	plan(fo)	geo	600
0.204	0.3178	2.2586	0.4544	0.7835	6.9637	plan(fo)	vect-geo	600
0.4049	0.2468	7.7277	0.7022	0.9253	13.4371	plan(fo)	vect	600
0.8861	0.4702	14.7883	1.7459	1.1721	21.4244	plan(f)	geo	650
0.268508	0.200103	5.257678	0.580974	0.720487	11.92539	plan(f)	vect-geo	650
0.6003	0.4318	11.8773	1.2444	1.0847	18.1697	plan(f)	vect	650
1.0114	0.4496	13.3892	1.9988	1.3601	17.0422	geom	geo	650
0.5682	0.2987	5.6961	1.1449	0.9056	9.0918	geom	vect-geo	650
0.7122	0.3539	10.5379	1.51	1.008	13.2564	geom	vect	650
1.3743	0.4717	19.2466	2.3208	1.417	22.9157	plan(squ)	geo	650
0.9416	0.2845	11.7336	1.5006	0.9288	15.1352	plan(squ)	vect-geo	650
1.0837	0.4282	16.4712	1.8446	1.2976	19.206	plan(squ)	vect	650
0.9048	0.4589	14.7849	1.7649	1.2259	21.4187	plan(fo)	geo	650
0.4646	0.263	6.9514	0.9392	0.7976	12.8961	plan(fo)	vect-geo	650
0.6176	0.4078	11.879	1.2646	1.0336	18.1744	plan(fo)	vect	650
0.7069	0.5205	15.1707	1.5075	1.2913	21.5108	plan(f)	geo	700
0.259747	0.210131	2.670706	0.61165	0.611713	6.58727	plan(f)	vect-geo	700
0.2511	0.5482	9.7673	0.6154	1.0969	15.862	plan(f)	vect	700
0.8086	0.4214	13.5528	1.7942	1.1737	17.4772	geom	geo	700
0.1046	0.125	1.3521	0.4123	0.3644	4.3387	geom	vect-geo	700
0.335	0.2948	8.2426	0.8617	0.9111	12.0803	geom	vect	700
1.2039	0.5288	19.7676	2.1619	1.5579	23.6941	plan(squ)	geo	700
0.4587	0.3029	6.2708	0.8095	0.6598	10.1017	plan(squ)	vect-geo	700

0.6961	0.5589	14.5351	1.2539	1.3127	18.3757	plan(squ)	vect	700
0.7242	0.499	15.164	1.5328	1.2466	21.5008	plan(fo)	geo	700
0.1057	0.2314	2.336	0.2939	0.6328	7.6567	plan(fo)	vect-geo	700
0.263	0.5054	9.7694	0.6355	1.0455	15.8674	plan(fo)	vect	700
0.7084	0.7219	17.0911	1.4144	1.5333	24.4879	plan(f)	geo	750
0.198281	0.48507	3.145637	0.697842	0.964113	8.411836	plan(f)	vect-geo	750
0.2992	0.7955	12.0462	0.7449	1.4441	18.3945	plan(f)	vect	750
0.7967	0.5124	15.2768	1.6092	1.283	18.3783	geom	geo	750
0.1564	0.234	2.5752	0.4946	0.5714	4.9943	geom	vect-geo	750
0.3769	0.4693	10.3323	0.952	1.1103	12.7854	geom	vect	750
1.2209	0.7489	21.8677	2.012	1.6869	25.0201	plan(squ)	geo	750
0.5494	0.604	9.1729	0.9187	0.9931	11.757	plan(squ)	vect-geo	750
0.7666	0.8293	17.0063	1.3673	1.5322	19.4474	plan(squ)	vect	750
0.7243	0.6885	17.085	1.4307	1.4703	24.4803	plan(fo)	geo	750
0.1235	0.5064	4.2459	0.4122	0.9808	9.6913	plan(fo)	vect-geo	750
0.3111	0.7491	12.0485	0.7651	1.3928	18.4005	plan(fo)	vect	750
1.4424	1.2026	35.1971	3.1603	2.891	45.5375	plan(f)	geo	800
0.268673	0.692332	4.092342	0.927592	1.26932	11.74073	plan(f)	vect-geo	800
0.5644	1.2454	22.5072	1.5407	2.3846	31.4082	plan(f)	vect	800
1.5178	1.0127	33.0835	3.39	2.7415	38.5918	geom	geo	800
0.2438	0.4265	3.9422	0.7531	1.1255	9.2887	geom	vect-geo	800
0.631	0.9146	20.5874	1.7823	2.168	25.1513	geom	vect	800
1.9596	1.2302	40.149	3.8386	3.1802	45.6989	plan(squ)	geo	800
0.5124	0.829	10.9842	1.2156	1.574	16.4705	plan(squ)	vect-geo	800
0.9648	1.2962	27.7274	2.2413	2.6205	32.3318	plan(squ)	vect	800
1.4577	1.1729	35.1814	3.1776	2.844	45.5173	plan(fo)	geo	800
0.2038	0.7102	5.5934	0.8813	1.3272	13.0459	plan(fo)	vect-geo	800
0.5732	1.1989	22.5102	1.561	2.3334	31.4151	plan(fo)	vect	800
1.5825	1.1453	40.8234	3.4942	2.9338	49.3501	plan(f)	geo	850
0.234288	0.460726	6.724728	0.925317	1.022145	12.71434	plan(f)	vect-geo	850
0.671	1.0356	27.0286	1.684	2.4028	34.0285	plan(f)	vect	850
1.6463	1.0263	38.4538	3.6984	2.6883	44.9559	geom	geo	850
0.3156	0.3641	6.8297	0.8569	1.0669	11.8878	geom	vect-geo	850
0.7317	0.7863	24.8799	1.9145	2.1479	30.0837	geom	vect	850
2.1157	1.1634	45.9186	4.1836	3.1601	52.4434	plan(squ)	geo	850
0.6258	0.587	14.3873	1.3586	1.5401	19.4521	plan(squ)	vect-geo	850
1.092	1.084	32.4222	2.413	2.6232	37.646	plan(squ)	vect	850
1.5976	1.1244	40.8065	3.5073	2.8822	49.3291	plan(fo)	geo	850
0.2672	0.5022	8.7588	0.8765	1.2406	14.2934	plan(fo)	vect-geo	850
0.6806	0.9979	27.0319	1.7043	2.3515	34.0342	plan(fo)	vect	850
1.757	1.1571	44.9389	3.9773	2.758	53.1882	plan(f)	geo	900
0.394663	0.276706	3.117862	0.923375	0.798016	8.878397	plan(f)	vect-geo	900
0.5842	0.9641	26.0711	1.5326	2.2114	33.7548	plan(f)	vect	900
1.8096	1.0619	42.2969	4.1685	2.5692	49.4466	geom	geo	900
0.2238	0.2538	1.9945	0.6894	0.64	6.6215	geom	vect-geo	900
0.6376	0.7153	23.7197	1.7598	1.877	29.9875	geom	vect	900
2.3303	1.1718	50.1424	4.6945	3.0581	57.3088	plan(squ)	geo	900
0.4366	0.4175	8.7184	1.0635	1.1338	14.5217	plan(squ)	vect-geo	900
1.0026	1.0296	31.6341	2.287	2.3749	37.9169	plan(squ)	vect	900
0.5931	0.9238	26.0748	1.5531	2.16	33.763	plan(fo)	geo	900
0.233	0.3191	3.6383	0.8721	0.9122	10.3928	plan(fo)	vect-geo	900
0.5931	0.9238	26.0748	1.5531	2.16	33.763	plan(fo)	vect	900

1.7976	0.9356	39.8023	3.4707	2.2543	51.3693	plan(f)	geo	950
0.171176	0.138178	4.003669	0.447054	0.40829	10.64881	plan(f)	vect-geo	950
0.6193	0.6869	22.1553	1.4064	1.7798	32.8204	plan(f)	vect	950
1.8498	0.8924	36.9653	3.6603	2.1721	44.6124	geom	geo	950
0.1673	0.3402	3.3632	0.78	0.7137	11.5357	geom	vect-geo	950
0.6759	0.4956	19.6097	1.6042	1.5225	28.0516	geom	vect	950
2.4443	0.9392	45.1369	4.2248	2.5441	52.81	plan(squ)	geo	950
0.7125	0.1951	5.929	1.3851	0.8662	15.1739	plan(squ)	vect-geo	950
1.2142	0.7475	27.8589	2.195	2.0428	36.3296	plan(squ)	vect	950
1.8146	0.925	39.7807	3.4865	2.2069	51.3423	plan(fo)	geo	950
0.1305	0.1684	3.2332	0.5526	0.5433	9.7277	plan(fo)	vect-geo	950
0.6335	0.6523	22.1593	1.4268	1.7284	32.8288	plan(fo)	vect	950
2.3997	1.1962	53.5179	4.9052	3.248	64.5124	plan(f)	geo	1000
0.412075	0.339559	7.116377	0.812854	0.987497	15.76436	plan(f)	vect-geo	1000
1.1297	0.8478	33.8641	2.425	2.3983	42.7964	plan(f)	vect	1000
2.4395	1.1717	50.4084	5.0674	3.0217	58.6662	geom	geo	1000
0.6203	0.4995	6.9338	1.2563	1.3085	14.0467	geom	vect-geo	1000
1.1793	0.7329	31.0857	2.6283	2.1522	38.3726	geom	vect	1000
3.0849	1.1912	59.0362	5.6751	3.5667	67.3401	plan(squ)	geo	1000
1.2498	0.3915	15.6238	1.8785	1.1683	22.7633	plan(squ)	vect-geo	1000
1.7797	0.8947	39.7921	3.2347	2.7027	47.1045	plan(squ)	vect	1000
2.4164	1.189	53.494	4.9168	3.1971	64.4842	plan(fo)	geo	1000
0.586	0.3931	9.4085	1.0833	1.0568	17.4629	plan(fo)	vect-geo	1000
1.1458	0.8285	33.8686	2.4456	2.3468	42.8054	plan(fo)	vect	1000
0.184	0.5324	8.6058	0.4557	1.0032	9.9685	plan(lin)	vect-geo	350
0.2064	0.5157	6.7339	0.466	0.8832	8.7607	plan(lin)	geo	350
0.1584	0.4735	7.3431	0.4274	0.9054	9.1634	plan(lin)	vect	350
0.2458	0.6483	10.0493	0.629	1.4645	12.9799	plan(lin)	vect-geo	400
0.3668	0.6919	12.5466	0.8931	1.71	15.4968	plan(lin)	geo	400
0.2955	0.7233	11.6618	0.7191	1.6515	14.5742	plan(lin)	vect	400
0.3128	0.3633	10.4736	1.0425	0.9924	12.9887	plan(lin)	vect-geo	450
0.8114	0.5875	17.6913	1.9093	1.6746	19.8286	plan(lin)	geo	450
0.4918	0.5538	14.9847	1.4301	1.494	17.0536	plan(lin)	vect	450
0.3545	0.2584	10.3485	0.9199	0.6195	12.5758	plan(lin)	vect-geo	500
0.9836	0.5381	19.6037	2.0499	1.3745	22.6921	plan(lin)	geo	500
0.5748	0.4318	16.0746	1.3848	1.1726	18.7634	plan(lin)	vect	500
0.4941	0.316	8.9742	1.1011	0.7167	11.6852	plan(lin)	vect-geo	550
1.0835	0.4521	18.1258	1.8941	1.2903	20.9877	plan(lin)	geo	550
0.6833	0.3069	14.6598	1.4466	0.8473	16.7873	plan(lin)	vect	550
0.5208	0.2764	7.7054	0.9089	0.6242	12.0223	plan(lin)	vect-geo	600
1.1269	0.4004	17.6406	1.7104	1.1915	22.0425	plan(lin)	geo	600
0.7157	0.2544	13.8686	1.1967	0.7638	18.155	plan(lin)	vect	600
0.7955	0.2821	13.7682	1.3587	0.9254	17.2308	plan(lin)	vect-geo	650
1.2257	0.47	21.3259	2.1785	1.4166	24.9507	plan(lin)	geo	650
0.9362	0.4257	18.5332	1.7051	1.2959	21.2973	plan(lin)	vect	650
0.2949	0.2979	8.5409	0.6501	0.656	12.2819	plan(lin)	vect-geo	700
1.0359	0.5259	22.1165	1.9996	1.5571	25.9961	plan(lin)	geo	700
0.53	0.5544	16.8521	1.0935	1.3105	20.6279	plan(lin)	vect	700
0.3652	0.5989	11.7246	0.7399	0.9894	14.2302	plan(lin)	vect-geo	750
1.0329	0.7449	24.4949	1.8301	1.6851	27.5855	plan(lin)	geo	750
0.5828	0.8246	19.6033	1.1857	1.5293	22.1264	plan(lin)	vect	750
0.3469	0.8238	13.8107	1.0176	1.5688	19.344	plan(lin)	vect-geo	800

1.7561	1.2268	43.1425	3.63	3.1789	48.6356	plan(lin)	geo	800
0.7985	1.2914	30.647	2.0383	2.6176	35.1851	plan(lin)	vect	800
0.4443	0.5824	17.4978	1.1402	1.5358	22.6443	plan(lin)	vect-geo	850
1.8961	1.1607	49.2097	3.954	3.1569	55.754	plan(lin)	geo	850
0.9094	1.0798	35.6331	2.1895	2.6188	40.8544	plan(lin)	vect	850
0.2537	0.4125	12.0637	0.8219	1.1265	17.888	plan(lin)	vect-geo	900
2.0882	1.1695	53.7234	4.4434	3.0561	60.937	plan(lin)	geo	900
0.8161	1.0252	35.1061	2.045	2.3712	41.41	plan(lin)	vect	900
0.4508	0.1911	9.3818	1.1229	0.8615	18.8744	plan(lin)	vect-geo	950
2.1706	0.9371	48.9579	3.9566	2.5415	56.7545	plan(lin)	geo	950
0.9553	0.7436	31.5775	1.9324	2.0389	40.1636	plan(lin)	vect	950
0.9649	0.3899	19.5389	1.5998	1.1618	26.6305	plan(lin)	vect-geo	1000
2.7867	1.1896	63.1993	5.3823	3.5642	71.6093	plan(lin)	geo	1000
1.4931	0.8917	43.8413	2.9514	2.6987	51.1299	plan(lin)	vect	1000

(1)plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).
(2)vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.
(3)All data is given in millimetres.

Table A2. Data set 2 of errors (mm) obtained using different camera calibration methods and different extraction techniques

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.2041	0.3156	3.282	0.7008	0.9089	8.5637	plan(f)	vect-geo	350
0.1145	0.141	3.0033	0.5074	0.7773	8.702	plan(f)	geo	350
0.1175	0.1803	3.3242	0.4867	0.8503	8.7045	plan(f)	vect	350
0.1134	0.1631	4.1474	0.4931	0.6835	9.8879	plan(f)	vect-geo	400
0.3157	0.3306	7.0156	0.8501	0.8735	12.2172	plan(f)	geo	400
0.2241	0.2879	6.2507	0.6042	0.7392	11.4461	plan(f)	vect	400
0.2269	0.5715	3.3343	0.4669	1.429	9.3886	plan(f)	vect-geo	450
1.1083	0.6076	14.482	2.0073	1.9626	20.4516	plan(f)	geo	450
0.5712	0.4264	10.2374	1.0989	1.4468	15.6145	plan(f)	vect	450
0.3336	0.7372	2.9308	0.5715	1.47	8.8024	plan(f)	vect-geo	500
1.3856	0.6919	16.9762	2.5801	2.3083	22.4985	plan(f)	geo	500
0.7099	0.4664	11.3509	1.4017	1.5922	16.8983	plan(f)	vect	500
0.4358	0.9796	2.6569	0.8865	1.5933	8.4914	plan(f)	vect-geo	550
1.4901	0.7886	16.7327	2.4676	2.2913	24.8572	plan(f)	geo	550
0.7948	0.5888	10.5542	1.4129	1.6779	18.0561	plan(f)	vect	550
0.3657	0.8393	2.9195	0.7274	1.3225	9.0717	plan(f)	vect-geo	600
1.3599	0.6324	14.5738	2.0539	2.0926	22.3146	plan(f)	geo	600
0.6804	0.4504	8.3167	1.1399	1.3649	14.9514	plan(f)	vect	600
0.4568	0.4588	2.6895	0.7435	0.9883	9.3318	plan(f)	vect-geo	650
1.3389	0.5007	16.4164	2.1748	1.6753	23.6568	plan(f)	geo	650
0.7567	0.3216	10.7689	1.1909	1.0735	17.9364	plan(f)	vect	650
0.2214	0.2156	3.3596	0.4887	0.6052	9.7955	plan(f)	vect-geo	700
1.1496	0.4457	16.9462	2.1022	1.1784	25.8579	plan(f)	geo	700
0.4994	0.2989	10.0747	1.0294	0.6083	19.3987	plan(f)	vect	700
0.1483	0.1454	2.7202	0.4337	0.468	8.2771	plan(f)	vect-geo	750
1.2118	0.6341	20.4911	2.6661	1.7681	32.1127	plan(f)	geo	750

0.5005	0.5507	12.2963	1.0856	1.2327	20.9565	plan(f)	vect	750
0.1623	0.4025	3.6828	0.5038	0.7658	12.9492	plan(f)	vect-geo	800
1.292	0.8723	25.8663	2.803	2.1794	34.3882	plan(f)	geo	800
0.5441	0.862	16.4241	1.4702	1.7332	24.9054	plan(f)	vect	800
0.3166	0.3292	7.5777	1.0674	0.9644	15.8943	plan(f)	vect-geo	850
1.9977	1.0948	43.2082	4.7743	2.8964	54.7499	plan(f)	geo	850
0.8637	0.8274	27.4891	2.4342	2.1674	37.2564	plan(f)	vect	850
0.1503	0.2427	3.6089	0.5672	0.8667	8.6013	plan(f)	vect-geo	900
2.3214	1.1783	50.0496	5.0756	3.0706	62.6618	plan(f)	geo	900
0.7098	0.7611	25.9286	2.0452	1.818	35.6924	plan(f)	vect	900
0.2425	0.325	2.8439	0.5623	1.1731	12.905	plan(f)	vect-geo	950
2.2785	1.0329	46.2142	4.6524	2.9839	63.829	plan(f)	geo	950
0.8684	0.6267	25.2964	2.0636	1.6843	39.8357	plan(f)	vect	950
0.5228	0.3462	5.679	0.9523	1.1944	18.6245	plan(f)	vect-geo	1000
2.6247	1.1934	54.8181	5.3013	2.883	69.2136	plan(f)	geo	1000
1.1558	0.7841	32.043	2.3672	1.8686	42.5294	plan(f)	vect	1000
0.1347	0.1684	3.9959	0.5202	0.8346	8.7484	plan(fo)	vect-geo	350
0.115	0.1395	3.002	0.4991	0.7608	8.688	plan(fo)	geo	350
0.1157	0.1715	3.323	0.4808	0.8325	8.6922	plan(fo)	vect	350
0.1691	0.2573	4.936	0.5513	0.6459	10.1263	plan(fo)	vect-geo	400
0.3113	0.3371	7.0137	0.8447	0.8867	12.1982	plan(fo)	geo	400
0.2212	0.2959	6.2493	0.6107	0.7573	11.4299	plan(fo)	vect	400
0.3035	0.4485	4.0094	0.6984	1.3467	9.5281	plan(fo)	vect-geo	450
1.1012	0.604	14.4777	2.0007	1.9452	20.4283	plan(fo)	geo	450
0.5658	0.4227	10.2356	1.0929	1.4293	15.6014	plan(fo)	vect	450
0.3945	0.6119	3.4646	0.7675	1.3832	9.0572	plan(fo)	vect-geo	500
1.3783	0.6866	16.9706	2.5715	2.2913	22.4739	plan(fo)	geo	500
0.7039	0.4585	11.3485	1.3955	1.5749	16.8805	plan(fo)	vect	500
0.4859	0.8575	2.6717	0.9142	1.5055	8.7677	plan(fo)	vect-geo	550
1.4825	0.7797	16.7262	2.459	2.2737	24.831	plan(fo)	geo	550
0.7883	0.5742	10.5513	1.4066	1.6606	18.0376	plan(fo)	vect	550
0.3916	0.7194	2.4427	0.683	1.2345	7.9384	plan(fo)	vect-geo	600
1.3524	0.6224	14.5665	2.0484	2.0754	22.2918	plan(fo)	geo	600
0.6734	0.4353	8.3132	1.1327	1.3477	14.9336	plan(fo)	vect	600
0.4903	0.3403	2.8873	0.7373	0.8978	9.6779	plan(fo)	vect-geo	650
1.3314	0.4975	16.4087	2.1684	1.6581	23.6296	plan(fo)	geo	650
0.7499	0.319	10.7648	1.184	1.0564	17.9164	plan(fo)	vect	650
0.2206	0.1335	3.144	0.4864	0.4793	10.1419	plan(fo)	vect-geo	700
1.1419	0.4488	16.9372	2.0949	1.1611	25.8307	plan(fo)	geo	700
0.4932	0.3062	10.07	1.0227	0.6244	19.3792	plan(fo)	vect	700
0.1648	0.2297	2.4495	0.4211	0.4551	8.6764	plan(fo)	vect-geo	750
1.2046	0.6396	20.4806	2.6572	1.7814	32.0849	plan(fo)	geo	750
0.4947	0.5636	12.291	1.079	1.2499	20.9388	plan(fo)	vect	750
0.1832	0.5064	4.0872	0.5615	0.9392	13.2992	plan(fo)	vect-geo	800
1.2849	0.8811	25.8544	2.7974	2.1928	34.3561	plan(fo)	geo	800
0.5389	0.8764	16.4182	1.4634	1.7502	24.8866	plan(fo)	vect	800
0.3555	0.3644	8.4498	1.1722	0.9824	16.5978	plan(fo)	vect-geo	850
1.9909	1.0981	43.1925	4.7684	2.9095	54.726	plan(fo)	geo	850
0.8589	0.8361	27.4824	2.4272	2.1844	37.2441	plan(fo)	vect	850
0.1508	0.2042	3.2872	0.6015	0.7697	8.1335	plan(fo)	vect-geo	900
2.3143	1.1806	50.0296	5.0688	3.0835	62.6174	plan(fo)	geo	900
0.7054	0.7697	25.9214	2.038	1.8347	35.6683	plan(fo)	vect	900

0.2677	0.2531	2.704	0.6532	1.0733	13.3278	plan(fo)	vect-geo	950
2.2712	1.0338	46.1945	4.6464	2.9659	63.788	plan(fo)	geo	950
0.8626	0.6338	25.2885	2.0562	1.701	39.813	plan(fo)	vect	950
0.5441	0.328	6.3958	1.0535	1.0945	19.2802	plan(fo)	vect-geo	1000
2.6174	1.1948	54.7967	5.2931	2.8649	69.1806	plan(fo)	geo	1000
1.1494	0.7901	32.0344	2.36	1.8852	42.5071	plan(fo)	vect	1000
0.3179	0.2043	8.0359	0.9048	0.8422	10.7913	plan(squ)	vect-geo	350
0.1953	0.1739	6.6728	0.7467	0.7732	9.7687	plan(squ)	geo	350
0.2654	0.1713	7.1652	0.8058	0.7176	10.1481	plan(squ)	vect	350
0.3609	0.3451	9.3104	0.9228	1.0206	11.6146	plan(squ)	vect-geo	400
0.4835	0.4181	10.9681	1.2079	1.306	13.6727	plan(squ)	geo	400
0.398	0.3764	10.3585	1.0615	1.1653	12.7328	plan(squ)	vect	400
0.5646	0.3531	8.6217	1.1055	1.0266	11.2588	plan(squ)	vect-geo	450
1.331	0.5968	18.6332	2.3814	1.6076	21.2872	plan(squ)	geo	450
0.8026	0.4252	14.593	1.5285	1.1048	17.044	plan(squ)	vect	450
0.6987	0.4521	8.2358	1.2257	0.9846	11.3802	plan(squ)	vect-geo	500
1.6389	0.6627	21.2973	2.9564	1.8716	24.4591	plan(squ)	geo	500
0.9807	0.444	15.9073	1.8607	1.1634	19.0358	plan(squ)	vect	500
0.818	0.6473	6.9234	1.325	1.1443	9.9722	plan(squ)	vect-geo	550
1.771	0.7012	21.1983	2.8743	1.9127	25.0259	plan(squ)	geo	550
1.1005	0.4751	15.2796	1.898	1.2447	18.8065	plan(squ)	vect	550
0.7355	0.4945	4.6696	1.0787	0.8935	8.8968	plan(squ)	vect-geo	600
1.6608	0.5386	19.1713	2.5572	1.6543	22.2937	plan(squ)	geo	600
1.0133	0.321	13.1918	1.6147	0.9848	16.4577	plan(squ)	vect	600
0.8649	0.1957	7.9412	1.1797	0.5471	10.9505	plan(squ)	vect-geo	650
1.6627	0.4987	21.1688	2.6733	1.2283	23.0837	plan(squ)	geo	650
1.1086	0.3458	15.8128	1.7558	0.9052	18.0577	plan(squ)	vect	650
0.5947	0.1774	5.8975	1.011	0.6116	10.9157	plan(squ)	vect-geo	700
1.4938	0.5207	21.8058	2.5521	1.3124	26.4438	plan(squ)	geo	700
0.8652	0.4593	15.2526	1.5484	1.002	19.4322	plan(squ)	vect	700
0.5658	0.4625	6.6828	0.9721	0.923	11.2219	plan(squ)	vect-geo	750
1.5725	0.768	25.4753	3.1678	2.2997	32.1627	plan(squ)	geo	750
0.8663	0.7852	17.6282	1.6706	1.7669	22.6711	plan(squ)	vect	750
0.5688	0.7835	9.4873	1.2948	1.4702	14.1612	plan(squ)	vect-geo	800
1.6643	1.0589	30.9778	3.3915	2.719	36.0891	plan(squ)	geo	800
0.8993	1.1311	21.9122	2.1485	2.2775	26.5287	plan(squ)	vect	800
0.726	0.4941	14.4597	1.7888	1.526	23.2384	plan(squ)	vect-geo	850
2.3832	1.1815	48.4324	5.385	3.4498	60.1319	plan(squ)	geo	850
1.2195	1.0043	33.1608	3.1401	2.7248	43.2233	plan(squ)	vect	850
0.4185	0.274	5.9611	1.2178	0.8611	11.671	plan(squ)	vect-geo	900
2.7307	1.2459	55.3252	5.6192	3.6328	66.2219	plan(squ)	geo	900
1.0672	0.9555	31.7305	2.6916	2.2898	37.9961	plan(squ)	vect	900
0.7439	0.231	7.5893	1.3456	0.8778	14.052	plan(squ)	vect-geo	950
2.7252	1.0786	51.5989	5.2914	3.1204	63.2145	plan(squ)	geo	950
1.2968	0.7995	31.2104	2.7736	2.2536	39.9643	plan(squ)	vect	950
1.0528	0.365	12.73	1.8415	1.133	21.8781	plan(squ)	vect-geo	1000
3.0993	1.252	60.3306	5.9225	3.3056	72.7351	plan(squ)	geo	1000
1.6181	0.9667	38.1191	3.1107	2.4471	44.9816	plan(squ)	vect	1000
0.1932	0.1625	3.2547	0.6719	0.6961	5.9632	geom	vect-geo	350
0.1575	0.1403	2.1275	0.5529	0.6481	5.242	geom	geo	350
0.1594	0.1464	2.5129	0.5984	0.6072	5.5109	geom	vect	350
0.2435	0.2743	4.0867	0.6646	0.8522	6.3872	geom	vect-geo	400

0.3379	0.3519	6.0541	1.0045	1.1592	8.7865	geom	geo	400
0.2785	0.3192	5.3268	0.8245	1.0308	7.7195	geom	vect	400
0.3193	0.3948	3.0322	0.8085	1.0485	5.6787	geom	vect-geo	450
1.074	0.5957	13.344	2.1433	1.654	16.0074	geom	geo	450
0.572	0.4083	9.1638	1.259	1.1294	11.6178	geom	vect	450
0.3856	0.5423	2.2811	0.9029	1.027	5.431	geom	vect-geo	500
1.3412	0.6807	15.6554	2.6741	1.929	18.827	geom	geo	500
0.6939	0.4431	10.1125	1.5574	1.1927	13.2472	geom	vect	500
0.4627	0.7854	1.3786	0.9737	1.2071	3.6213	geom	vect-geo	550
1.438	0.7603	15.2185	2.5634	2.0017	19.0016	geom	geo	550
0.7611	0.5346	9.1364	1.5663	1.3064	12.6204	geom	vect	550
0.3537	0.6529	2.2771	0.6991	0.9772	6.2715	geom	vect-geo	600
1.3001	0.6065	12.8651	2.2408	1.7639	15.9333	geom	geo	600
0.6363	0.3936	6.7125	1.257	1.073	9.9816	geom	vect	600
0.4515	0.2891	1.1562	0.7741	0.6536	3.8994	geom	vect-geo	650
1.2718	0.4866	14.5054	2.319	1.3352	16.4224	geom	geo	650
0.7063	0.3028	8.9673	1.3626	0.7259	11.2216	geom	vect	650
0.1722	0.1039	2.6319	0.57	0.3511	7.5419	geom	vect-geo	700
1.074	0.4361	14.8029	2.1551	1.0483	19.4168	geom	geo	700
0.4547	0.3041	8.0554	1.1299	0.756	12.2088	geom	vect	700
0.1542	0.2474	1.5541	0.5005	0.6236	6.0642	geom	vect-geo	750
1.1324	0.6249	18.1126	2.7385	2.0035	24.785	geom	geo	750
0.4599	0.5806	10.0584	1.2217	1.4752	15.0924	geom	vect	750
0.1895	0.5377	1.9156	0.8004	1.1296	5.9268	geom	vect-geo	800
1.2101	0.8687	23.2424	2.953	2.4001	28.2949	geom	geo	800
0.5144	0.8965	13.9569	1.6678	1.9627	18.5156	geom	vect	800
0.3679	0.3462	5.887	1.2602	1.1602	14.6186	geom	vect-geo	850
1.9098	1.073	40.2624	4.9108	3.1037	51.8937	geom	geo	850
0.8348	0.8212	24.7532	2.6206	2.3813	34.7466	geom	vect	850
0.2247	0.1767	3.3005	0.6605	0.5785	7.0369	geom	vect-geo	900
2.2208	1.1523	46.7977	5.1011	3.2662	57.648	geom	geo	900
0.6807	0.7491	22.956	2.1365	1.945	29.1764	geom	vect	900
0.2254	0.2428	2.4333	0.7603	0.8136	6.9686	geom	vect-geo	950
2.1656	1.003	42.7456	4.7607	2.7335	54.2743	geom	geo	950
0.8017	0.6061	22.0976	2.1973	1.8682	30.7638	geom	vect	950
0.4717	0.3099	3.6146	1.2277	0.8452	12.1433	geom	vect-geo	1000
2.5045	1.161	51.0842	5.343	2.9017	63.4386	geom	geo	1000
1.0714	0.757	28.6011	2.5051	2.0389	35.4112	geom	vect	1000
0.3022	0.2034	8.3561	0.8909	0.838	11.1558	plan(lin)	vect-geo	350
0.1812	0.1727	6.9883	0.7322	0.7717	10.1523	plan(lin)	geo	350
0.2499	0.1717	7.4825	0.7917	0.7152	10.5248	plan(lin)	vect	350
0.3384	0.3379	9.8424	0.8976	1.0121	12.2006	plan(lin)	vect-geo	400
0.457	0.4114	11.5089	1.1778	1.3001	14.2809	plan(lin)	geo	400
0.3752	0.3691	10.8959	1.0341	1.1585	13.3271	plan(lin)	vect	400
0.5188	0.3613	9.3569	1.061	1.0394	12.0514	plan(lin)	vect-geo	450
1.2824	0.5992	19.4134	2.3364	1.6202	22.0895	plan(lin)	geo	450
0.7586	0.4267	15.3542	1.4866	1.1178	17.8196	plan(lin)	vect	450
0.6372	0.4675	9.1756	1.1693	1.001	12.3794	plan(lin)	vect-geo	500
1.5745	0.668	22.2958	2.8969	1.8869	25.5271	plan(lin)	geo	500
0.9197	0.4469	16.8807	1.8026	1.1794	20.0743	plan(lin)	vect	500
0.7434	0.6691	8.0639	1.255	1.1643	11.1353	plan(lin)	vect-geo	550
1.6933	0.7119	22.4034	2.8013	1.9331	26.2369	plan(lin)	geo	550

1.0242	0.4864	16.457	1.8264	1.2644	19.9971	plan(lin)	vect	550
0.6479	0.5198	6.007	0.9917	0.9171	10.2846	plan(lin)	vect-geo	600
1.5702	0.5508	20.5751	2.4708	1.6783	23.6536	plan(lin)	geo	600
0.9241	0.3334	14.5673	1.5311	1.0088	17.9009	plan(lin)	vect	600
0.7631	0.2088	9.4986	1.0826	0.5746	12.5503	plan(lin)	vect-geo	650
1.5582	0.4987	22.7881	2.5731	1.2424	24.7349	plan(lin)	geo	650
1.0054	0.3398	17.4061	1.6564	0.8778	19.6965	plan(lin)	vect	650
0.4799	0.1538	7.6464	0.8984	0.5797	12.7377	plan(lin)	vect-geo	700
1.3758	0.5083	23.635	2.4356	1.2806	28.3297	plan(lin)	geo	700
0.7487	0.4323	17.05	1.4356	0.9696	21.2746	plan(lin)	vect	700
0.4372	0.427	8.6469	0.8453	0.8871	13.251	plan(lin)	vect-geo	750
1.4401	0.7432	27.5271	3.0381	2.2646	34.2967	plan(lin)	geo	750
0.7376	0.7506	19.6422	1.5447	1.7314	24.7579	plan(lin)	vect	750
0.4272	0.7431	11.6692	1.1573	1.4298	16.3236	plan(lin)	vect-geo	800
1.5184	1.029	33.2599	3.2476	2.6798	38.3664	plan(lin)	geo	800
0.7683	1.0904	24.1506	2.008	2.2379	28.7691	plan(lin)	vect	800
0.5854	0.4672	16.868	1.6357	1.4812	25.7207	plan(lin)	vect-geo	850
2.2286	1.1638	50.9947	5.2221	3.4055	62.7808	plan(lin)	geo	850
1.0866	0.9728	35.6518	2.9827	2.6804	45.8002	plan(lin)	vect	850
0.2734	0.245	8.5397	1.0503	0.8125	14.3386	plan(lin)	vect-geo	900
2.5578	1.2307	58.124	5.4388	3.5843	69.0916	plan(lin)	geo	900
0.9281	0.9193	34.4213	2.519	2.2407	40.7355	plan(lin)	vect	900
0.5604	0.2186	10.3813	1.1649	0.8258	16.8104	plan(lin)	vect-geo	950
2.5325	1.0643	54.5889	5.1017	3.0684	66.1983	plan(lin)	geo	950
1.1193	0.7644	34.1062	2.5901	2.2018	42.8386	plan(lin)	vect	950
0.8544	0.3473	15.7497	1.6482	1.077	24.8775	plan(lin)	vect-geo	1000
2.891	1.2357	63.5645	5.7163	3.2481	76.004	plan(lin)	geo	1000
1.4276	0.9292	41.2503	2.9131	2.3912	48.1069	plan(lin)	vect	1000

(1)plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).

(2)vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.

(3)All data is given in millimetres.

Table A3. Data set 1 of errors (mm) obtained using different camera calibration methods and different extraction techniques and only *x*-axis data

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method[1]	Error type[2]	Dist.
0.288	0.6323	5.2896	0.9029	1.126	9.2364	plan(f)	vect-geo	350
0.293	0.6581	5.2908	0.9216	1.1904	9.234	plan(f)	geo	350
0.2841	0.6072	5.2885	0.8841	1.0624	9.2387	plan(f)	vect	350
0.3036	0.6992	5.2053	0.8445	1.2914	9.7181	plan(f)	vect-geo	400
0.3072	0.6271	5.2047	0.8544	1.2244	9.7209	plan(f)	geo	400
0.3005	0.7722	5.2059	0.8346	1.3585	9.7153	plan(f)	vect	400
0.2539	0.3213	4.143	0.7251	0.7442	9.4288	plan(f)	vect-geo	450
0.2591	0.1975	4.1409	0.7424	0.6179	9.4389	plan(f)	geo	450
0.2486	0.4749	4.1452	0.7078	0.8704	9.4187	plan(f)	vect	450
0.2034	0.1148	3.1559	0.5507	0.3883	7.879	plan(f)	vect-geo	500
0.2087	0.1911	3.1538	0.5672	0.5596	7.8929	plan(f)	geo	500
0.1996	0.2233	3.1579	0.5343	0.4959	7.8652	plan(f)	vect	500
0.1161	0.3194	2.0303	0.3654	0.5795	6.181	plan(f)	vect-geo	550
0.1287	0.4988	2.0327	0.3562	0.7561	6.189	plan(f)	geo	550
0.1066	0.1456	2.028	0.3746	0.416	6.173	plan(f)	vect	550
0.1731	0.3068	2.112	0.6809	0.6225	5.424	plan(f)	vect-geo	600
0.1807	0.488	2.1158	0.6735	0.8264	5.4404	plan(f)	geo	600
0.1658	0.1384	2.1081	0.6882	0.4187	5.4076	plan(f)	vect	600
0.2954	0.1826	5.3109	0.6988	0.5125	11.367	plan(f)	vect-geo	650
0.2942	0.174	5.3091	0.6968	0.6359	11.3686	plan(f)	geo	650
0.2972	0.2736	5.3126	0.7009	0.61	11.3655	plan(f)	vect	650
0.3125	0.2388	2.7203	0.8677	0.4717	6.9701	plan(f)	vect-geo	700
0.3174	0.1288	2.7251	0.8492	0.4369	6.9856	plan(f)	geo	700
0.3094	0.431	2.7155	0.8862	0.6874	6.9547	plan(f)	vect	700
0.2328	0.5111	2.5779	0.9087	0.8788	7.1297	plan(f)	vect-geo	750
0.2364	0.3298	2.5804	0.8998	0.7447	7.1412	plan(f)	geo	750
0.2302	0.6985	2.5755	0.9177	1.0128	7.1181	plan(f)	vect	750
0.5143	0.6691	2.4041	0.9743	0.9846	6.9421	plan(f)	vect-geo	800
0.5152	0.2951	2.4126	0.9552	0.6555	6.9592	plan(f)	geo	800
0.5134	1.0485	2.3956	0.9933	1.3137	6.9251	plan(f)	vect	800
0.4475	0.3778	3.2473	0.8508	0.682	7.7466	plan(f)	vect-geo	850
0.4475	0.1197	3.2503	0.8385	0.3423	7.7586	plan(f)	geo	850
0.4475	0.7646	3.2444	0.8631	1.0217	7.7346	plan(f)	vect	850
0.7529	0.2523	4.842	1.4761	0.5532	11.3641	plan(f)	vect-geo	900
0.7522	0.2892	4.8477	1.4823	0.9758	11.3997	plan(f)	geo	900
0.7535	0.6608	4.8363	1.47	1.0213	11.3285	plan(f)	vect	900
0.4633	0.2316	7.6545	1.2342	0.7689	16.8593	plan(f)	vect-geo	950
0.4618	0.4785	7.66	1.2315	1.1841	16.8712	plan(f)	geo	950
0.4656	0.4327	7.6495	1.237	0.9441	16.8474	plan(f)	vect	950
0.1264	0.1435	3.2325	0.479	0.3797	9.2669	plan(f)	vect-geo	1000
0.1319	0.5615	3.2415	0.468	0.8527	9.2743	plan(f)	geo	1000
0.1216	0.3416	3.2234	0.4901	0.6815	9.2595	plan(f)	vect	1000
0.2728	0.589	5.2909	0.8793	1.0696	9.2501	plan(fo)	vect-geo	350
0.2771	0.6171	5.2921	0.8941	1.1283	9.2479	plan(fo)	geo	350
0.2697	0.5611	5.2897	0.8645	1.011	9.2523	plan(fo)	vect	350
0.2862	0.6545	5.207	0.8228	1.2451	9.7322	plan(fo)	vect-geo	400
0.2887	0.5856	5.2064	0.8305	1.1835	9.7351	plan(fo)	geo	400
0.2844	0.7234	5.2076	0.815	1.3066	9.7293	plan(fo)	vect	400

0.2337	0.2822	4.1451	0.7018	0.6868	9.4442	plan(fo)	vect-geo	450
0.2379	0.1789	4.1429	0.7154	0.5541	9.4543	plan(fo)	geo	450
0.2297	0.4265	4.1474	0.6881	0.8196	9.4341	plan(fo)	vect	450
0.1842	0.1149	3.1587	0.5275	0.4408	7.8953	plan(fo)	vect-geo	500
0.1888	0.2289	3.1566	0.5405	0.6138	7.9091	plan(fo)	geo	500
0.1811	0.1825	3.1608	0.5145	0.4451	7.8815	plan(fo)	vect	500
0.1128	0.3686	2.0349	0.3478	0.6323	6.1972	plan(fo)	vect-geo	550
0.1215	0.5458	2.0374	0.3409	0.8029	6.2054	plan(fo)	geo	550
0.1057	0.1914	2.0325	0.3548	0.4685	6.189	plan(fo)	vect	550
0.1761	0.3563	2.1158	0.6631	0.6689	5.4262	plan(fo)	vect-geo	600
0.1822	0.5359	2.1198	0.6579	0.8677	5.4426	plan(fo)	geo	600
0.17	0.1785	2.1118	0.6684	0.4701	5.4098	plan(fo)	vect	600
0.315	0.168	5.3156	0.7189	0.5663	11.3846	plan(fo)	vect-geo	650
0.3134	0.1801	5.3139	0.7172	0.6909	11.3866	plan(fo)	geo	650
0.3167	0.2322	5.3173	0.7207	0.5585	11.3827	plan(fo)	vect	650
0.2934	0.2013	2.7226	0.8518	0.414	6.9696	plan(fo)	vect-geo	700
0.2974	0.124	2.7276	0.8375	0.4797	6.9849	plan(fo)	geo	700
0.2899	0.3805	2.7177	0.866	0.6361	6.9543	plan(fo)	vect	700
0.2137	0.4618	2.5846	0.8911	0.8188	7.1505	plan(fo)	vect-geo	750
0.217	0.284	2.5872	0.8846	0.676	7.1621	plan(fo)	geo	750
0.2112	0.6472	2.582	0.8976	0.9616	7.1389	plan(fo)	vect	750
0.4939	0.6178	2.4094	0.9584	0.9242	6.9623	plan(fo)	vect-geo	800
0.4948	0.2474	2.418	0.9439	0.5858	6.9793	plan(fo)	geo	800
0.4931	0.9971	2.4008	0.9729	1.2625	6.9452	plan(fo)	vect	800
0.4272	0.326	3.254	0.8334	0.6336	7.7628	plan(fo)	vect-geo	850
0.4273	0.1313	3.2568	0.8241	0.3042	7.7747	plan(fo)	geo	850
0.4271	0.7132	3.2511	0.8427	0.9702	7.7509	plan(fo)	vect	850
0.7327	0.22	4.8362	1.4555	0.5493	11.3661	plan(fo)	vect-geo	900
0.7323	0.3385	4.8422	1.4612	1.027	11.4023	plan(fo)	geo	900
0.733	0.6117	4.8302	1.4499	0.9692	11.3299	plan(fo)	vect	900
0.4446	0.2332	7.6476	1.2154	0.819	16.8505	plan(fo)	vect-geo	950
0.4426	0.5288	7.6536	1.214	1.2328	16.8625	plan(fo)	geo	950
0.4468	0.3933	7.6419	1.2168	0.8916	16.8384	plan(fo)	vect	950
0.1316	0.1787	3.2386	0.4619	0.4373	9.2895	plan(fo)	vect-geo	1000
0.1357	0.6159	3.2476	0.4541	0.9008	9.2971	plan(fo)	geo	1000
0.1279	0.2903	3.2296	0.4697	0.6303	9.282	plan(fo)	vect	1000
0.2314	0.5477	8.6046	0.648	1.0723	10.1989	plan(squ)	vect-geo	350
0.2317	0.5958	8.6049	0.6492	1.1034	10.1942	plan(squ)	geo	350
0.2311	0.5012	8.6044	0.6469	1.0413	10.2036	plan(squ)	vect	350
0.2041	0.6354	8.8234	0.5938	1.3368	11.7127	plan(squ)	vect-geo	400
0.2041	0.579	8.8236	0.5941	1.2805	11.7113	plan(squ)	geo	400
0.2041	0.6926	8.8231	0.5936	1.393	11.714	plan(squ)	vect	400
0.1757	0.2705	8.0247	0.7083	0.6697	10.807	plan(squ)	vect-geo	450
0.1756	0.1848	8.0251	0.7064	0.5212	10.818	plan(squ)	geo	450
0.1758	0.4134	8.0243	0.7102	0.8182	10.7961	plan(squ)	vect	450
0.2156	0.1216	7.2354	0.5873	0.2961	9.4281	plan(squ)	vect-geo	500
0.2154	0.206	7.2362	0.5864	0.4773	9.4441	plan(squ)	geo	500
0.2157	0.1729	7.2345	0.5881	0.4645	9.412	plan(squ)	vect	500
0.4092	0.3312	5.6162	0.8506	0.4725	8.7402	plan(squ)	vect-geo	550
0.4091	0.4957	5.6174	0.8495	0.6606	8.7326	plan(squ)	geo	550
0.4093	0.1681	5.615	0.8518	0.288	8.7478	plan(squ)	vect	550
0.4538	0.303	3.9319	0.7427	0.4883	8.2069	plan(squ)	vect-geo	600

0.4538	0.4709	3.9334	0.7433	0.6671	8.2045	plan(squ)	geo	600
0.4538	0.1395	3.9304	0.742	0.3095	8.2093	plan(squ)	vect	600
0.7995	0.1904	10.1577	1.2889	0.6796	14.1174	plan(squ)	vect-geo	650
0.7995	0.1765	10.1591	1.2886	0.5853	14.1162	plan(squ)	geo	650
0.7995	0.2713	10.1563	1.2892	0.7861	14.1185	plan(squ)	vect	650
0.2437	0.2536	3.568	0.5461	0.5398	7.341	plan(squ)	vect-geo	700
0.2438	0.1013	3.5707	0.5462	0.3104	7.3568	plan(squ)	geo	700
0.2436	0.4555	3.5652	0.5461	0.7692	7.3252	plan(squ)	vect	700
0.3409	0.56	6.5701	0.5036	0.8007	9.5089	plan(squ)	vect-geo	750
0.3409	0.3819	6.5728	0.5044	0.635	9.5236	plan(squ)	geo	750
0.3408	0.741	6.5675	0.5028	0.9663	9.4942	plan(squ)	vect	750
0.1562	0.7353	5.4435	0.4164	0.9817	11.5522	plan(squ)	vect-geo	800
0.1564	0.3632	5.4504	0.417	0.6211	11.5548	plan(squ)	geo	800
0.1559	1.1074	5.4367	0.4159	1.4127	11.5496	plan(squ)	vect	800
0.1911	0.4598	8.4195	0.6154	0.9291	14.3767	plan(squ)	vect-geo	850
0.1911	0.1125	8.4273	0.6158	0.6062	14.3733	plan(squ)	geo	850
0.191	0.841	8.4117	0.6149	1.252	14.3802	plan(squ)	vect	850
0.2177	0.2993	2.1132	0.7241	0.6709	7.6445	plan(squ)	vect-geo	900
0.2181	0.2126	2.1197	0.726	0.7462	7.6568	plan(squ)	geo	900
0.2173	0.7504	2.1066	0.7221	1.1359	7.6322	plan(squ)	vect	900
0.3345	0.209	3.0049	0.7876	0.7044	11.5562	plan(squ)	vect-geo	950
0.3349	0.3847	3.0051	0.7875	0.9059	11.5511	plan(squ)	geo	950
0.334	0.5028	3.0047	0.7878	1.1458	11.5613	plan(squ)	vect	950
0.7198	0.0882	7.6356	1.0589	0.3612	14.6951	plan(squ)	vect-geo	1000
0.7199	0.4335	7.6474	1.06	0.6827	14.7298	plan(squ)	geo	1000
0.7198	0.4671	7.6238	1.0578	0.7777	14.6604	plan(squ)	vect	1000
0.2382	0.3554	4.8005	0.6225	0.8232	6.4045	geom	vect-geo	350
0.2393	0.3966	4.7996	0.6179	0.8624	6.3989	geom	geo	350
0.2377	0.3165	4.8014	0.6272	0.784	6.4102	geom	vect	350
0.2368	0.4017	4.6398	0.6381	1.0633	7.546	geom	vect-geo	400
0.2371	0.3546	4.6408	0.6338	1.0147	7.5443	geom	geo	400
0.2365	0.4519	4.6387	0.6424	1.1119	7.5478	geom	vect	400
0.1652	0.1566	3.4692	0.5115	0.3816	6.2526	geom	vect-geo	450
0.1666	0.2049	3.4725	0.5069	0.5253	6.2663	geom	geo	450
0.164	0.1819	3.4659	0.5162	0.521	6.2389	geom	vect	450
0.113	0.3019	2.3637	0.3392	0.561	4.5066	geom	vect-geo	500
0.1155	0.4778	2.3683	0.3457	0.7568	4.5268	geom	geo	500
0.1112	0.1525	2.3592	0.3327	0.3733	4.4864	geom	vect	500
0.129	0.641	1.1002	0.5745	0.8014	3.4148	geom	vect-geo	550
0.1312	0.8053	1.101	0.5733	0.9887	3.4094	geom	geo	550
0.1272	0.4767	1.0995	0.5757	0.614	3.4202	geom	vect	550
0.1634	0.635	1.9611	0.4092	0.832	4.8894	geom	vect-geo	600
0.1652	0.8034	1.9568	0.4226	1.026	4.8974	geom	geo	600
0.1619	0.4667	1.9655	0.4038	0.6551	4.8813	geom	vect	600
0.424	0.2566	4.0963	0.9258	0.6648	8.0325	geom	vect-geo	650
0.425	0.3603	4.1006	0.9263	0.8032	8.0327	geom	geo	650
0.4229	0.1801	4.092	0.9252	0.5265	8.0324	geom	vect	650
0.1939	0.1462	2.9225	0.7312	0.4082	7.4729	geom	vect-geo	700
0.195	0.3293	2.9149	0.7402	0.6152	7.4749	geom	geo	700
0.1933	0.1167	2.9301	0.7221	0.4247	7.4709	geom	vect	700
0.1215	0.1712	1.2858	0.6654	0.3791	5.2004	geom	vect-geo	750
0.1217	0.083	1.2881	0.6712	0.5257	5.2076	geom	geo	750

0.1217	0.3443	1.2836	0.6595	0.5541	5.1932	geom	vect	750
0.4072	0.3103	2.4342	0.8656	0.588	7.1332	geom	vect-geo	800
0.4064	0.1129	2.4253	0.8749	0.3768	7.1282	geom	geo	800
0.408	0.6855	2.443	0.8562	1.0285	7.1382	geom	vect	800
0.3528	0.0839	2.0698	0.7442	0.454	6.789	geom	vect-geo	850
0.3529	0.3798	2.0796	0.75	0.5789	6.7953	geom	geo	850
0.3527	0.3957	2.0601	0.7384	0.7736	6.7827	geom	vect	850
0.6637	0.223	6.7509	1.262	0.77	13.0747	geom	vect-geo	900
0.6634	0.6604	6.7244	1.261	1.2462	13.0456	geom	geo	900
0.664	0.307	6.7774	1.263	0.6987	13.1037	geom	vect	900
0.4045	0.4423	9.7668	1.038	1.0076	19.7602	geom	vect-geo	950
0.4062	0.8606	9.741	1.0411	1.4293	19.7371	geom	geo	950
0.4032	0.2201	9.7927	1.0349	0.6976	19.7832	geom	vect	950
0.1354	0.5014	2.835	0.4075	0.7192	7.7986	geom	vect-geo	1000
0.1372	0.9583	2.8314	0.4254	1.1991	7.7824	geom	geo	1000
0.1336	0.0894	2.8386	0.3897	0.3051	7.8148	geom	vect	1000
0.2316	0.5428	9.027	0.6384	1.0703	10.4843	plan(lin)	vect-geo	350
0.2319	0.5912	9.0272	0.6395	1.1011	10.4796	plan(lin)	geo	350
0.2313	0.4963	9.0269	0.6374	1.0396	10.489	plan(lin)	vect	350
0.205	0.6303	9.5124	0.5529	1.3341	12.4506	plan(lin)	vect-geo	400
0.2051	0.5743	9.5127	0.5536	1.2775	12.4492	plan(lin)	geo	400
0.2049	0.6873	9.5121	0.5521	1.3907	12.4519	plan(lin)	vect	400
0.1525	0.2659	8.975	0.6569	0.6673	11.6271	plan(lin)	vect-geo	450
0.1525	0.1825	8.9756	0.655	0.5189	11.6383	plan(lin)	geo	450
0.1524	0.4081	8.9744	0.6589	0.8158	11.6159	plan(lin)	vect	450
0.1444	0.1219	8.4471	0.5092	0.2992	10.5127	plan(lin)	vect-geo	500
0.1444	0.21	8.4482	0.5087	0.4801	10.523	plan(lin)	geo	500
0.1444	0.1684	8.4459	0.5098	0.4616	10.5156	plan(lin)	vect	500
0.3052	0.3364	7.0852	0.7568	0.4786	10.3554	plan(lin)	vect-geo	550
0.3053	0.5008	7.0867	0.7556	0.667	10.348	plan(lin)	geo	550
0.3052	0.1732	7.0838	0.7579	0.2914	10.3627	plan(lin)	vect	550
0.3378	0.3083	5.6576	0.6196	0.4936	9.9885	plan(lin)	vect-geo	600
0.3379	0.476	5.6594	0.6211	0.6725	9.9864	plan(lin)	geo	600
0.3377	0.1444	5.6558	0.6181	0.3148	9.9906	plan(lin)	vect	600
0.6547	0.1875	12.183	1.1485	0.6756	16.2038	plan(lin)	vect-geo	650
0.6547	0.1758	12.1846	1.1482	0.5813	16.2028	plan(lin)	geo	650
0.6547	0.2667	12.1813	1.1488	0.7816	16.2048	plan(lin)	vect	650
0.1452	0.2483	5.7807	0.5007	0.5365	9.4969	plan(lin)	vect-geo	700
0.1456	0.0999	5.7838	0.5023	0.3076	9.5131	plan(lin)	geo	700
0.1448	0.4501	5.7776	0.4991	0.7655	9.4807	plan(lin)	vect	700
0.174	0.5545	9.1065	0.4165	0.7925	11.9167	plan(lin)	vect-geo	750
0.1742	0.3766	9.1095	0.4185	0.6262	11.9318	plan(lin)	geo	750
0.1739	0.7356	9.1034	0.4145	0.9589	11.9017	plan(lin)	vect	750
0.1572	0.7298	8.2385	0.5975	0.9791	14.3944	plan(lin)	vect-geo	800
0.1577	0.3576	8.2462	0.5994	0.6149	14.3978	plan(lin)	geo	800
0.1568	1.102	8.2309	0.5957	1.41	14.3911	plan(lin)	vect	800
0.1127	0.4543	11.496	0.454	0.924	17.5345	plan(lin)	vect-geo	850
0.1132	0.1089	11.5047	0.4552	0.601	17.5318	plan(lin)	geo	850
0.1122	0.8356	11.4873	0.4528	1.247	17.5372	plan(lin)	vect	850
0.3498	0.2944	4.5114	0.9536	0.6673	10.9644	plan(lin)	vect-geo	900
0.3499	0.2173	4.5226	0.9554	0.754	10.9777	plan(lin)	geo	900
0.3497	0.745	4.5001	0.9519	1.132	10.951	plan(lin)	vect	900

0.2708	0.209	3.1497	0.715	0.7019	11.8442	plan(lin)	vect-geo	950
0.2712	0.3899	3.1568	0.7171	0.9138	11.8426	plan(lin)	geo	950
0.2703	0.498	3.1425	0.7129	1.1433	11.8458	plan(lin)	vect	950
0.4411	0.0877	11.5054	0.7773	0.3537	18.5741	plan(lin)	vect-geo	1000
0.4411	0.4394	11.5184	0.7791	0.6874	18.5825	plan(lin)	geo	1000
0.441	0.4617	11.4924	0.7755	0.7753	18.5657	plan(lin)	vect	1000

(1)plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).

(2)vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.

(3)All data is given in millimetres.

Table A4. Data set 2 of errors (mm) obtained using different camera calibration methods and different extraction techniques and only *x*-axis data

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.1721	0.1911	4.2349	0.5463	0.8859	9.1321	plan(f)	vect-geo	350
0.1742	0.1963	4.225	0.5685	0.8527	9.1158	plan(f)	geo	350
0.1735	0.2067	4.2447	0.5242	0.9192	9.1483	plan(f)	vect	350
0.1301	0.2171	4.4856	0.5049	0.5416	9.6846	plan(f)	vect-geo	400
0.1293	0.2154	4.5001	0.4825	0.5953	9.7032	plan(f)	geo	400
0.1342	0.2216	4.471	0.5273	0.5734	9.6659	plan(f)	vect	400
0.1098	0.5133	2.2733	0.3006	1.0585	7.5968	plan(f)	vect-geo	450
0.1118	0.6996	2.3136	0.3152	1.2812	7.6719	plan(f)	geo	450
0.113	0.327	2.2352	0.3128	0.8357	7.5216	plan(f)	vect	450
0.1647	0.6899	1.7994	0.4659	0.9548	6.3631	plan(f)	vect-geo	500
0.1734	0.9182	1.8304	0.4684	1.2305	6.4797	plan(f)	geo	500
0.1582	0.4616	1.7715	0.4635	0.6821	6.2465	plan(f)	vect	500
0.2575	0.9358	2.5193	0.5667	1.3146	6.4115	plan(f)	vect-geo	550
0.2622	1.1709	2.4815	0.5216	1.5524	6.3047	plan(f)	geo	550
0.2554	0.7007	2.5582	0.6118	1.0768	6.5182	plan(f)	vect	550
0.2847	0.7972	3.9053	0.9314	1.5195	11.6181	plan(f)	vect-geo	600
0.2824	1.024	3.8297	0.9275	1.7418	11.499	plan(f)	geo	600
0.2893	0.5703	3.9841	0.9353	1.2973	11.7373	plan(f)	vect	600
0.2773	0.4108	2.4145	0.7625	0.7267	7.7977	plan(f)	vect-geo	650
0.2841	0.6074	2.3948	0.7605	0.9118	7.7039	plan(f)	geo	650
0.2705	0.2147	2.4362	0.7645	0.5416	7.8915	plan(f)	vect	650
0.2351	0.213	4.0315	1.1744	0.8602	11.7802	plan(f)	vect-geo	700
0.2289	0.4004	3.9563	1.1759	1.0942	11.645	plan(f)	geo	700
0.2423	0.1959	4.1076	1.1728	0.6695	11.9155	plan(f)	vect	700
0.2106	0.1641	3.6367	0.7539	0.5403	8.1608	plan(f)	vect-geo	750
0.2028	0.1924	3.5485	0.7528	0.6904	8.0091	plan(f)	geo	750
0.222	0.3612	3.7294	0.7551	0.8104	8.3125	plan(f)	vect	750
0.221	0.3988	2.9469	0.7897	0.7306	9.3193	plan(f)	vect-geo	800
0.21	0.1632	2.915	0.7776	0.4919	9.4763	plan(f)	geo	800
0.2349	0.6717	2.9812	0.8064	0.98	9.1623	plan(f)	vect	800
0.2531	0.1217	3.9398	1.0614	0.5754	10.5424	plan(f)	vect-geo	850
0.2427	0.4253	4.0092	1.0412	1.0259	10.7503	plan(f)	geo	850
0.2666	0.3757	3.8793	1.0815	0.7093	10.3346	plan(f)	vect	850
0.6946	0.3352	9.1388	1.7199	1.4826	16.7186	plan(f)	vect-geo	900
0.6785	0.7915	8.7974	1.6831	2.0305	16.3649	plan(f)	geo	900
0.7108	0.3851	9.4801	1.7567	0.9815	17.0722	plan(f)	vect	900

0.3524	0.3664	7.1309	1.2553	1.0416	13.9729	plan(f)	vect-geo	950
0.339	0.813	6.8514	1.2331	1.4861	13.6696	plan(f)	geo	950
0.3668	0.2454	7.4104	1.2776	0.7029	14.2762	plan(f)	vect	950
0.2003	0.2782	4.2172	0.9299	0.748	11.4059	plan(f)	vect-geo	1000
0.1959	0.7554	4.0803	0.9099	1.2361	11.7287	plan(f)	geo	1000
0.2096	0.2333	4.3751	0.95	0.6179	11.0831	plan(f)	vect	1000
0.168	0.1896	4.2345	0.5388	0.8692	9.1231	plan(fo)	vect-geo	350
0.1694	0.1968	4.2249	0.5597	0.8366	9.1071	plan(fo)	geo	350
0.17	0.2002	4.2441	0.5179	0.9018	9.1391	plan(fo)	vect	350
0.1296	0.2252	4.4845	0.5129	0.5396	9.6739	plan(fo)	vect-geo	400
0.1284	0.2213	4.4989	0.4918	0.5783	9.6923	plan(fo)	geo	400
0.1342	0.2322	4.4701	0.534	0.5894	9.6555	plan(fo)	vect	400
0.1092	0.4977	2.2704	0.3074	1.0413	7.5873	plan(fo)	vect-geo	450
0.11	0.6854	2.3102	0.3039	1.2641	7.6612	plan(fo)	geo	450
0.1124	0.3099	2.2328	0.3193	0.8185	7.5134	plan(fo)	vect	450
0.1607	0.6742	1.7959	0.4721	0.9376	6.35	plan(fo)	vect-geo	500
0.1684	0.9039	1.8262	0.4739	1.2133	6.465	plan(fo)	geo	500
0.1552	0.4446	1.7683	0.4702	0.6651	6.2351	plan(fo)	vect	500
0.2535	0.9201	2.5174	0.5619	1.2997	6.4123	plan(fo)	vect-geo	550
0.2573	1.1565	2.4797	0.5196	1.5399	6.3075	plan(fo)	geo	550
0.2518	0.6838	2.5556	0.6042	1.0594	6.5171	plan(fo)	vect	550
0.2821	0.7815	3.906	0.9378	1.504	11.6175	plan(fo)	vect-geo	600
0.2796	1.0095	3.8315	0.9335	1.7279	11.5004	plan(fo)	geo	600
0.2863	0.5535	3.9837	0.9421	1.28	11.7345	plan(fo)	vect	600
0.2715	0.3951	2.4115	0.7689	0.7111	7.7963	plan(fo)	vect-geo	650
0.2782	0.5928	2.3922	0.7664	0.8978	7.7042	plan(fo)	geo	650
0.2649	0.1984	2.433	0.7713	0.5244	7.8883	plan(fo)	vect	650
0.2345	0.2031	4.0328	1.1806	0.8452	11.7817	plan(fo)	vect-geo	700
0.2282	0.386	3.9585	1.1814	1.0791	11.6486	plan(fo)	geo	700
0.2415	0.2025	4.1076	1.1798	0.6524	11.9147	plan(fo)	vect	700
0.2139	0.1719	3.6377	0.7604	0.5565	8.1626	plan(fo)	vect-geo	750
0.2058	0.1821	3.551	0.7587	0.6752	8.0133	plan(fo)	geo	750
0.2244	0.3762	3.7293	0.762	0.827	8.3119	plan(fo)	vect	750
0.2274	0.4141	2.9441	0.7966	0.7443	9.3047	plan(fo)	vect-geo	800
0.2156	0.1727	2.9128	0.7846	0.5029	9.4592	plan(fo)	geo	800
0.2406	0.6882	2.9784	0.8133	0.9965	9.1502	plan(fo)	vect	800
0.2592	0.1209	3.9352	1.0685	0.56	10.5291	plan(fo)	vect-geo	850
0.2486	0.4106	4.0032	1.0485	1.0118	10.7337	plan(fo)	geo	850
0.2723	0.3916	3.8758	1.0885	0.7247	10.3246	plan(fo)	vect	850
0.7022	0.3256	9.1489	1.7277	1.4678	16.7211	plan(fo)	vect-geo	900
0.6863	0.7765	8.8131	1.6917	2.0176	16.3732	plan(fo)	geo	900
0.7181	0.3941	9.4847	1.7637	0.9978	17.0689	plan(fo)	vect	900
0.357	0.353	7.1392	1.2626	1.0263	13.9792	plan(fo)	vect-geo	950
0.3433	0.7978	6.8643	1.2404	1.4719	13.6813	plan(fo)	geo	950
0.3713	0.252	7.4141	1.2847	0.7191	14.2772	plan(fo)	vect	950
0.2	0.2632	4.2192	0.9372	0.7327	11.3883	plan(fo)	vect-geo	1000
0.1951	0.7402	4.0843	0.9171	1.2219	11.7062	plan(fo)	geo	1000
0.2086	0.2448	4.375	0.9572	0.6336	11.0705	plan(fo)	vect	1000
0.3801	0.2275	8.4263	1.0468	0.9018	11.0999	plan(squ)	vect-geo	350
0.3846	0.2529	8.4137	1.0374	0.9669	11.0851	plan(squ)	geo	350
0.3761	0.2227	8.4389	1.0562	0.8367	11.1147	plan(squ)	vect	350
0.3181	0.3195	8.9498	0.913	0.8911	11.5733	plan(squ)	vect-geo	400

0.3248	0.3152	8.9615	0.9335	0.8758	11.5705	plan(squ)	geo	400
0.3114	0.3259	8.9381	0.8925	0.9063	11.5761	plan(squ)	vect	400
0.3238	0.3263	6.5679	0.6803	0.7297	9.3261	plan(squ)	vect-geo	450
0.3354	0.4981	6.6414	0.7265	0.9484	9.3869	plan(squ)	geo	450
0.3122	0.1765	6.4945	0.6341	0.511	9.2653	plan(squ)	vect	450
0.4068	0.4807	5.5985	0.6772	0.7623	8.7594	plan(squ)	vect-geo	500
0.4189	0.7004	5.6937	0.7248	1.0397	8.8433	plan(squ)	geo	500
0.3948	0.2616	5.5032	0.6295	0.4848	8.6754	plan(squ)	vect	500
0.5243	0.7113	4.05	0.8035	0.9417	7.0781	plan(squ)	vect-geo	550
0.5356	0.9375	4.1546	0.7917	1.2057	7.145	plan(squ)	geo	550
0.5129	0.4851	3.9453	0.8153	0.7102	7.0112	plan(squ)	vect	550
0.4648	0.5578	1.991	0.8189	1.0857	6.3705	plan(squ)	vect-geo	600
0.4749	0.7755	2.0706	0.8419	1.3097	6.4553	plan(squ)	geo	600
0.4548	0.3453	1.9127	0.7959	0.8617	6.2858	plan(squ)	vect	600
0.6126	0.1578	5.2313	0.7948	0.358	8.5085	plan(squ)	vect-geo	650
0.622	0.3416	5.33	0.8322	0.5698	8.5916	plan(squ)	geo	650
0.6033	0.0633	5.1327	0.7574	0.248	8.4254	plan(squ)	vect	650
0.3486	0.1886	3.2746	0.6685	0.5408	8.74	plan(squ)	vect-geo	700
0.3567	0.1828	3.332	0.7103	0.7062	8.8115	plan(squ)	geo	700
0.3408	0.3276	3.2266	0.6438	0.7943	8.6685	plan(squ)	vect	700
0.2611	0.3868	3.1412	0.5198	0.884	7.2951	plan(squ)	vect-geo	750
0.2699	0.1785	3.2556	0.5414	0.6216	7.4288	plan(squ)	geo	750
0.2527	0.6293	3.0275	0.4982	1.1463	7.1614	plan(squ)	vect	750
0.2549	0.6999	5.213	0.5106	0.997	10.2989	plan(squ)	vect-geo	800
0.2647	0.4359	5.3718	0.5728	0.7329	10.451	plan(squ)	geo	800
0.2468	0.9639	5.0542	0.4794	1.2611	10.1467	plan(squ)	vect	800
0.2549	0.295	8.0399	0.7471	0.7878	17.7787	plan(squ)	vect-geo	850
0.2673	0.1636	8.2819	0.7806	0.5321	17.9715	plan(squ)	geo	850
0.2433	0.6793	7.7979	0.7135	1.087	17.5859	plan(squ)	vect	850
0.3178	0.2448	3.3658	1.0174	0.9262	6.9737	plan(squ)	vect-geo	900
0.3076	0.4756	3.1183	0.9811	1.4726	6.6316	plan(squ)	geo	900
0.3297	0.6131	3.6228	1.0538	1.3118	7.3158	plan(squ)	vect	900
0.2788	0.1803	2.0641	0.6928	0.5893	6.3741	plan(squ)	vect-geo	950
0.2872	0.4637	2.0142	0.7024	0.9749	6.0399	plan(squ)	geo	950
0.2712	0.4595	2.1446	0.6833	1.0623	6.7083	plan(squ)	vect	950
0.4791	0.1345	4.3902	0.8019	0.3825	14.1422	plan(squ)	vect-geo	1000
0.4919	0.3849	4.6362	0.8453	0.7143	14.4591	plan(squ)	geo	1000
0.4663	0.559	4.1471	0.7927	0.8424	13.8253	plan(squ)	vect	1000
0.2326	0.1819	3.5118	0.7939	0.7381	6.1694	geom	vect-geo	350
0.2345	0.1965	3.5022	0.7867	0.7851	6.1572	geom	geo	350
0.2316	0.1829	3.5213	0.8011	0.6912	6.1816	geom	vect	350
0.2129	0.2421	3.6596	0.6305	0.7065	6.2822	geom	vect-geo	400
0.2143	0.2307	3.674	0.6457	0.6737	6.2823	geom	geo	400
0.2115	0.26	3.6452	0.6152	0.7393	6.282	geom	vect	400
0.0847	0.4172	1.2554	0.3703	0.7408	3.6951	geom	vect-geo	450
0.0916	0.6085	1.2921	0.4064	0.9676	3.758	geom	geo	450
0.0802	0.2298	1.223	0.3342	0.514	3.6322	geom	vect	450
0.0932	0.5981	1.067	0.3387	0.8728	2.7557	geom	vect-geo	500
0.1024	0.8307	1.0228	0.3764	1.1523	2.8413	geom	geo	500
0.0857	0.3656	1.1171	0.301	0.5933	2.6701	geom	vect	500
0.1843	0.8487	2.4231	0.4417	1.133	4.8624	geom	vect-geo	550
0.1916	1.0874	2.3406	0.4358	1.381	4.7309	geom	geo	550

0.1785	0.61	2.507	0.4475	0.885	4.994	geom	vect	550
0.1932	0.7155	4.986	0.7482	1.3369	9.8435	geom	vect-geo	600
0.1927	0.9453	4.8778	0.7322	1.5743	9.7282	geom	geo	600
0.1937	0.4874	5.0943	0.7641	1.0996	9.9588	geom	vect	600
0.2069	0.3351	1.9374	0.4474	0.5952	4.8759	geom	vect-geo	650
0.2154	0.5342	1.8445	0.4423	0.8207	4.7866	geom	geo	650
0.1994	0.1445	2.032	0.4525	0.3697	4.9652	geom	vect	650
0.1738	0.1943	4.861	0.9643	0.7219	11.345	geom	vect-geo	700
0.1715	0.3469	4.7543	0.9607	0.9926	11.2133	geom	geo	700
0.1778	0.2205	4.9702	0.9679	0.7065	11.4767	geom	vect	700
0.2219	0.1968	4.9321	0.5608	0.7565	10.1973	geom	vect-geo	750
0.2144	0.1675	4.7914	0.5337	0.599	10.0261	geom	geo	750
0.2303	0.4177	5.0727	0.5878	1.0284	10.3685	geom	vect	750
0.2726	0.4546	3.1958	0.7144	0.8374	7.7799	geom	vect-geo	800
0.2643	0.1942	3.0683	0.7048	0.5617	7.63	geom	geo	800
0.2823	0.729	3.325	0.7241	1.1131	7.9298	geom	vect	800
0.3234	0.1189	3.2565	0.8609	0.4808	9.0934	geom	vect-geo	850
0.3143	0.3779	3.207	0.8398	0.9025	9.2869	geom	geo	850
0.3338	0.4222	3.313	0.8821	0.7954	8.8998	geom	vect	850
0.7762	0.3372	11.7855	1.5781	1.3109	15.8474	geom	vect-geo	900
0.76	0.7479	11.45	1.5431	1.8703	15.5075	geom	geo	900
0.7924	0.3938	12.1209	1.6131	1.1312	16.1874	geom	vect	900
0.4109	0.3494	9.8222	1.0878	0.9326	15.6659	geom	vect-geo	950
0.3974	0.7728	9.5152	1.0641	1.3838	15.3343	geom	geo	950
0.4245	0.2568	10.1291	1.1116	0.8608	15.9975	geom	vect	950
0.1767	0.2487	5.8435	0.7769	0.6501	13.3709	geom	vect-geo	1000
0.1704	0.7213	5.555	0.7551	1.144	13.029	geom	geo	1000
0.185	0.2462	6.1507	0.7988	0.6095	13.7129	geom	vect	1000
0.3644	0.2261	8.7485	1.0386	0.8962	11.5213	plan(lin)	vect-geo	350
0.3685	0.2502	8.7358	1.0291	0.9608	11.5062	plan(lin)	geo	350
0.3604	0.2228	8.7611	1.0481	0.8316	11.5364	plan(lin)	vect	350
0.2959	0.3108	9.4799	0.885	0.8814	12.1766	plan(lin)	vect-geo	400
0.3037	0.3071	9.4917	0.9055	0.8657	12.1737	plan(lin)	geo	400
0.2901	0.3168	9.4681	0.8644	0.897	12.1794	plan(lin)	vect	400
0.277	0.3398	7.294	0.6363	0.7426	10.0993	plan(lin)	vect-geo	450
0.2886	0.5125	7.3679	0.6826	0.961	10.1607	plan(lin)	geo	450
0.2654	0.1866	7.2201	0.59	0.5242	10.0379	plan(lin)	vect	450
0.3468	0.4988	6.5265	0.6196	0.7812	9.7448	plan(lin)	vect-geo	500
0.3588	0.7184	6.6224	0.6673	1.0591	9.8295	plan(lin)	geo	500
0.3347	0.2791	6.4307	0.5719	0.5034	9.6602	plan(lin)	vect	500
0.4514	0.7329	5.1777	0.7296	0.9626	8.2722	plan(lin)	vect-geo	550
0.463	0.9591	5.283	0.7177	1.2285	8.3396	plan(lin)	geo	550
0.4399	0.5067	5.0725	0.7415	0.731	8.2048	plan(lin)	vect	550
0.3847	0.5829	3.1187	0.7307	1.1114	7.7523	plan(lin)	vect-geo	600
0.3946	0.8006	3.2225	0.7537	1.3358	7.8375	plan(lin)	geo	600
0.3753	0.3689	3.0183	0.7076	0.8871	7.6672	plan(lin)	vect	600
0.5152	0.186	6.7762	0.6949	0.388	10.0949	plan(lin)	vect-geo	650
0.5245	0.3707	6.8754	0.7324	0.6001	10.1786	plan(lin)	geo	650
0.5058	0.0591	6.6769	0.6574	0.2202	10.0111	plan(lin)	vect	650
0.2619	0.173	4.5372	0.6241	0.5106	10.5495	plan(lin)	vect-geo	700
0.2694	0.2007	4.6422	0.6251	0.7417	10.6216	plan(lin)	geo	700
0.2571	0.3003	4.4351	0.6231	0.7639	10.4774	plan(lin)	vect	700

0.1707	0.3518	4.8899	0.3912	0.8496	9.3041	plan(lin)	vect-geo	750
0.1769	0.1597	5.0198	0.4128	0.5875	9.4388	plan(lin)	geo	750
0.1664	0.5931	4.7634	0.3695	1.1116	9.1693	plan(lin)	vect	750
0.1616	0.6599	7.3753	0.3701	0.9584	12.4468	plan(lin)	vect-geo	800
0.1703	0.3958	7.535	0.4325	0.6946	12.5996	plan(lin)	geo	800
0.1548	0.9239	7.2155	0.356	1.2223	12.294	plan(lin)	vect	800
0.1517	0.2547	10.4195	0.5914	0.7436	20.2356	plan(lin)	vect-geo	850
0.1599	0.1886	10.6629	0.6249	0.5774	20.4297	plan(lin)	geo	850
0.1461	0.6355	10.1761	0.5579	1.0428	20.0414	plan(lin)	vect	850
0.4072	0.2429	2.0353	1.1776	0.9734	6.1094	plan(lin)	vect-geo	900
0.395	0.5164	1.9664	1.1413	1.5202	6.3725	plan(lin)	geo	900
0.4202	0.5694	2.1464	1.2139	1.2668	5.8462	plan(lin)	vect	900
0.2233	0.1879	2.8959	0.67	0.586	8.3682	plan(lin)	vect-geo	950
0.2253	0.5119	3.0901	0.6507	1.0272	8.6195	plan(lin)	geo	950
0.2243	0.4106	2.7122	0.6923	1.0137	8.1169	plan(lin)	vect	950
0.305	0.1169	7.0381	0.607	0.3254	17.1034	plan(lin)	vect-geo	1000
0.3146	0.4397	7.3512	0.6514	0.771	17.422	plan(lin)	geo	1000
0.2966	0.5045	6.7251	0.5978	0.7893	16.7848	plan(lin)	vect	1000

(1) plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).

(2) vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.

(3) All data is given in millimetres.

Table A5. Data set 1 of errors (mm) obtained using different camera calibration methods and different extraction techniques and only *x*-axis data (f_x and f_y recalculated for planes methods).

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.288	0.6323	5.2896	0.9029	1.126	9.2364	plan(f)	vect-geo	350
0.293	0.6581	5.2908	0.9216	1.1904	9.234	plan(f)	geo	350
0.2841	0.6072	5.2885	0.8841	1.0624	9.2387	plan(f)	vect	350
0.3036	0.6992	5.2053	0.8445	1.2914	9.7181	plan(f)	vect-geo	400
0.3072	0.6271	5.2047	0.8544	1.2244	9.7209	plan(f)	geo	400
0.3005	0.7722	5.2059	0.8346	1.3585	9.7153	plan(f)	vect	400
0.2539	0.3213	4.143	0.7251	0.7442	9.4288	plan(f)	vect-geo	450
0.2591	0.1975	4.1409	0.7424	0.6179	9.4389	plan(f)	geo	450
0.2486	0.4749	4.1452	0.7078	0.8704	9.4187	plan(f)	vect	450
0.2034	0.1148	3.1559	0.5507	0.3883	7.879	plan(f)	vect-geo	500
0.2087	0.1911	3.1538	0.5672	0.5596	7.8929	plan(f)	geo	500
0.1996	0.2233	3.1579	0.5343	0.4959	7.8652	plan(f)	vect	500
0.1161	0.3194	2.0303	0.3654	0.5795	6.181	plan(f)	vect-geo	550
0.1287	0.4988	2.0327	0.3562	0.7561	6.189	plan(f)	geo	550
0.1066	0.1456	2.028	0.3746	0.416	6.173	plan(f)	vect	550
0.1731	0.3068	2.112	0.6809	0.6225	5.424	plan(f)	vect-geo	600
0.1807	0.488	2.1158	0.6735	0.8264	5.4404	plan(f)	geo	600
0.1658	0.1384	2.1081	0.6882	0.4187	5.4076	plan(f)	vect	600
0.2954	0.1826	5.3109	0.6988	0.5125	11.367	plan(f)	vect-geo	650
0.2942	0.174	5.3091	0.6968	0.6359	11.3686	plan(f)	geo	650
0.2972	0.2736	5.3126	0.7009	0.61	11.3655	plan(f)	vect	650
0.3125	0.2388	2.7203	0.8677	0.4717	6.9701	plan(f)	vect-geo	700
0.3174	0.1288	2.7251	0.8492	0.4369	6.9856	plan(f)	geo	700
0.3094	0.431	2.7155	0.8862	0.6874	6.9547	plan(f)	vect	700
0.2328	0.5111	2.5779	0.9087	0.8788	7.1297	plan(f)	vect-geo	750
0.2364	0.3298	2.5804	0.8998	0.7447	7.1412	plan(f)	geo	750
0.2302	0.6985	2.5755	0.9177	1.0128	7.1181	plan(f)	vect	750
0.5143	0.6691	2.4041	0.9743	0.9846	6.9421	plan(f)	vect-geo	800
0.5152	0.2951	2.4126	0.9552	0.6555	6.9592	plan(f)	geo	800
0.5134	1.0485	2.3956	0.9933	1.3137	6.9251	plan(f)	vect	800
0.4475	0.3778	3.2473	0.8508	0.682	7.7466	plan(f)	vect-geo	850
0.4475	0.1197	3.2503	0.8385	0.3423	7.7586	plan(f)	geo	850
0.4475	0.7646	3.2444	0.8631	1.0217	7.7346	plan(f)	vect	850
0.7529	0.2523	4.842	1.4761	0.5532	11.3641	plan(f)	vect-geo	900
0.7522	0.2892	4.8477	1.4823	0.9758	11.3997	plan(f)	geo	900
0.7535	0.6608	4.8363	1.47	1.0213	11.3285	plan(f)	vect	900
0.4633	0.2316	7.6545	1.2342	0.7689	16.8593	plan(f)	vect-geo	950
0.4618	0.4785	7.66	1.2315	1.1841	16.8712	plan(f)	geo	950
0.4656	0.4327	7.6495	1.237	0.9441	16.8474	plan(f)	vect	950
0.1264	0.1435	3.2325	0.479	0.3797	9.2669	plan(f)	vect-geo	1000
0.1319	0.5615	3.2415	0.468	0.8527	9.2743	plan(f)	geo	1000
0.1216	0.3416	3.2234	0.4901	0.6815	9.2595	plan(f)	vect	1000
0.2728	0.589	5.2909	0.8793	1.0696	9.2501	plan(fo)	vect-geo	350
0.2771	0.6171	5.2921	0.8941	1.1283	9.2479	plan(fo)	geo	350
0.2697	0.5611	5.2897	0.8645	1.011	9.2523	plan(fo)	vect	350
0.2862	0.6545	5.207	0.8228	1.2451	9.7322	plan(fo)	vect-geo	400

0.2887	0.5856	5.2064	0.8305	1.1835	9.7351	plan(fo)	geo	400
0.2844	0.7234	5.2076	0.815	1.3066	9.7293	plan(fo)	vect	400
0.2337	0.2822	4.1451	0.7018	0.6868	9.4442	plan(fo)	vect-geo	450
0.2379	0.1789	4.1429	0.7154	0.5541	9.4543	plan(fo)	geo	450
0.2297	0.4265	4.1474	0.6881	0.8196	9.4341	plan(fo)	vect	450
0.1842	0.1149	3.1587	0.5275	0.4408	7.8953	plan(fo)	vect-geo	500
0.1888	0.2289	3.1566	0.5405	0.6138	7.9091	plan(fo)	geo	500
0.1811	0.1825	3.1608	0.5145	0.4451	7.8815	plan(fo)	vect	500
0.1128	0.3686	2.0349	0.3478	0.6323	6.1972	plan(fo)	vect-geo	550
0.1215	0.5458	2.0374	0.3409	0.8029	6.2054	plan(fo)	geo	550
0.1057	0.1914	2.0325	0.3548	0.4685	6.189	plan(fo)	vect	550
0.1761	0.3563	2.1158	0.6631	0.6689	5.4262	plan(fo)	vect-geo	600
0.1822	0.5359	2.1198	0.6579	0.8677	5.4426	plan(fo)	geo	600
0.17	0.1785	2.1118	0.6684	0.4701	5.4098	plan(fo)	vect	600
0.315	0.168	5.3156	0.7189	0.5663	11.3846	plan(fo)	vect-geo	650
0.3134	0.1801	5.3139	0.7172	0.6909	11.3866	plan(fo)	geo	650
0.3167	0.2322	5.3173	0.7207	0.5585	11.3827	plan(fo)	vect	650
0.2934	0.2013	2.7226	0.8518	0.414	6.9696	plan(fo)	vect-geo	700
0.2974	0.124	2.7276	0.8375	0.4797	6.9849	plan(fo)	geo	700
0.2899	0.3805	2.7177	0.866	0.6361	6.9543	plan(fo)	vect	700
0.2137	0.4618	2.5846	0.8911	0.8188	7.1505	plan(fo)	vect-geo	750
0.217	0.284	2.5872	0.8846	0.676	7.1621	plan(fo)	geo	750
0.2112	0.6472	2.582	0.8976	0.9616	7.1389	plan(fo)	vect	750
0.4939	0.6178	2.4094	0.9584	0.9242	6.9623	plan(fo)	vect-geo	800
0.4948	0.2474	2.418	0.9439	0.5858	6.9793	plan(fo)	geo	800
0.4931	0.9971	2.4008	0.9729	1.2625	6.9452	plan(fo)	vect	800
0.4272	0.326	3.254	0.8334	0.6336	7.7628	plan(fo)	vect-geo	850
0.4273	0.1313	3.2568	0.8241	0.3042	7.7747	plan(fo)	geo	850
0.4271	0.7132	3.2511	0.8427	0.9702	7.7509	plan(fo)	vect	850
0.7327	0.22	4.8362	1.4555	0.5493	11.3661	plan(fo)	vect-geo	900
0.7323	0.3385	4.8422	1.4612	1.027	11.4023	plan(fo)	geo	900
0.733	0.6117	4.8302	1.4499	0.9692	11.3299	plan(fo)	vect	900
0.4446	0.2332	7.6476	1.2154	0.819	16.8505	plan(fo)	vect-geo	950
0.4426	0.5288	7.6536	1.214	1.2328	16.8625	plan(fo)	geo	950
0.4468	0.3933	7.6419	1.2168	0.8916	16.8384	plan(fo)	vect	950
0.1316	0.1787	3.2386	0.4619	0.4373	9.2895	plan(fo)	vect-geo	1000
0.1357	0.6159	3.2476	0.4541	0.9008	9.2971	plan(fo)	geo	1000
0.1279	0.2903	3.2296	0.4697	0.6303	9.282	plan(fo)	vect	1000
0.2499	0.4706	4.7855	0.6611	0.9157	6.434	plan(squ)	vect-geo	350
0.2501	0.5115	4.7857	0.6608	0.9555	6.4298	plan(squ)	geo	350
0.2498	0.4309	4.7852	0.6613	0.8913	6.4382	plan(squ)	vect	350
0.2473	0.5527	4.671	0.6566	1.1746	7.5425	plan(squ)	vect-geo	400
0.2472	0.4903	4.6712	0.6564	1.1117	7.5411	plan(squ)	geo	400
0.2474	0.6168	4.6708	0.6568	1.2375	7.544	plan(squ)	vect	400
0.1693	0.1905	3.5459	0.4619	0.5172	6.3744	plan(squ)	vect-geo	450
0.1691	0.1419	3.5464	0.462	0.3796	6.3859	plan(squ)	geo	450
0.1694	0.3328	3.5455	0.4619	0.6723	6.3628	plan(squ)	vect	450
0.1118	0.1316	2.4765	0.333	0.3347	4.6726	plan(squ)	vect-geo	500
0.1118	0.2901	2.4777	0.3337	0.531	4.6893	plan(squ)	geo	500
0.1117	0.1047	2.4753	0.3323	0.314	4.656	plan(squ)	vect	500
0.1422	0.4283	1.1248	0.5617	0.6162	3.5378	plan(squ)	vect-geo	550
0.1422	0.5995	1.1271	0.5604	0.8108	3.5297	plan(squ)	geo	550

0.1422	0.257	1.1225	0.5629	0.4215	3.5458	plan(squ)	vect	550
0.1908	0.4052	1.8064	0.4405	0.6282	4.7279	plan(squ)	vect-geo	600
0.1911	0.5798	1.8053	0.4412	0.8164	4.7405	plan(squ)	geo	600
0.1906	0.2329	1.8076	0.4398	0.4401	4.7154	plan(squ)	vect	600
0.4778	0.1444	4.3424	0.9638	0.506	8.2622	plan(squ)	vect-geo	650
0.4777	0.1737	4.3438	0.9634	0.5063	8.261	plan(squ)	geo	650
0.4778	0.1827	4.341	0.9642	0.607	8.2634	plan(squ)	vect	650
0.1562	0.1644	2.6532	0.6847	0.495	7.1999	plan(squ)	vect-geo	700
0.1564	0.1198	2.6507	0.686	0.4039	7.2087	plan(squ)	geo	700
0.156	0.3488	2.6558	0.6834	0.7312	7.191	plan(squ)	vect	700
0.0843	0.4429	1.3263	0.6068	0.6507	4.9041	plan(squ)	vect-geo	750
0.0845	0.2649	1.3291	0.6087	0.4785	4.9177	plan(squ)	geo	750
0.0841	0.6284	1.3235	0.6049	0.8229	4.8906	plan(squ)	vect	750
0.2981	0.6106	2.2293	0.7788	0.9334	6.7551	plan(squ)	vect-geo	800
0.2982	0.2359	2.2268	0.7804	0.4957	6.7617	plan(squ)	geo	800
0.298	0.9892	2.2318	0.7772	1.371	6.7486	plan(squ)	vect	800
0.2267	0.3298	2.2433	0.6322	0.7344	7.1938	plan(squ)	vect-geo	850
0.2267	0.1043	2.249	0.6333	0.4051	7.1903	plan(squ)	geo	850
0.2266	0.7175	2.2376	0.6312	1.0637	7.1972	plan(squ)	vect	850
0.5155	0.2084	6.2764	1.1348	0.5969	12.5836	plan(squ)	vect-geo	900
0.5156	0.3355	6.2661	1.1368	0.9546	12.5719	plan(squ)	geo	900
0.5155	0.6225	6.2867	1.1328	1.0684	12.5953	plan(squ)	vect	900
0.3154	0.2417	9.257	0.897	0.7021	19.2414	plan(squ)	vect-geo	950
0.3156	0.5193	9.2463	0.8991	1.1193	19.2362	plan(squ)	geo	950
0.3152	0.4035	9.2677	0.8948	1.089	19.2466	plan(squ)	vect	950
0.2774	0.1439	2.7048	0.6082	0.3533	7.2372	plan(squ)	vect-geo	1000
0.2776	0.5852	2.7082	0.6093	0.8394	7.238	plan(squ)	geo	1000
0.2772	0.3282	2.7014	0.6071	0.7155	7.2363	plan(squ)	vect	1000
0.2382	0.3554	4.8005	0.6225	0.8232	6.4045	geom	vect-geo	350
0.2393	0.3966	4.7996	0.6179	0.8624	6.3989	geom	geo	350
0.2377	0.3165	4.8014	0.6272	0.784	6.4102	geom	vect	350
0.2368	0.4017	4.6398	0.6381	1.0633	7.546	geom	vect-geo	400
0.2371	0.3546	4.6408	0.6338	1.0147	7.5443	geom	geo	400
0.2365	0.4519	4.6387	0.6424	1.1119	7.5478	geom	vect	400
0.1652	0.1566	3.4692	0.5115	0.3816	6.2526	geom	vect-geo	450
0.1666	0.2049	3.4725	0.5069	0.5253	6.2663	geom	geo	450
0.164	0.1819	3.4659	0.5162	0.521	6.2389	geom	vect	450
0.113	0.3019	2.3637	0.3392	0.561	4.5066	geom	vect-geo	500
0.1155	0.4778	2.3683	0.3457	0.7568	4.5268	geom	geo	500
0.1112	0.1525	2.3592	0.3327	0.3733	4.4864	geom	vect	500
0.129	0.641	1.1002	0.5745	0.8014	3.4148	geom	vect-geo	550
0.1312	0.8053	1.101	0.5733	0.9887	3.4094	geom	geo	550
0.1272	0.4767	1.0995	0.5757	0.614	3.4202	geom	vect	550
0.1634	0.635	1.9611	0.4092	0.832	4.8894	geom	vect-geo	600
0.1652	0.8034	1.9568	0.4226	1.026	4.8974	geom	geo	600
0.1619	0.4667	1.9655	0.4038	0.6551	4.8813	geom	vect	600
0.424	0.2566	4.0963	0.9258	0.6648	8.0325	geom	vect-geo	650
0.425	0.3603	4.1006	0.9263	0.8032	8.0327	geom	geo	650
0.4229	0.1801	4.092	0.9252	0.5265	8.0324	geom	vect	650
0.1939	0.1462	2.9225	0.7312	0.4082	7.4729	geom	vect-geo	700
0.195	0.3293	2.9149	0.7402	0.6152	7.4749	geom	geo	700
0.1933	0.1167	2.9301	0.7221	0.4247	7.4709	geom	vect	700

0.1215	0.1712	1.2858	0.6654	0.3791	5.2004	geom	vect-geo	750
0.1217	0.083	1.2881	0.6712	0.5257	5.2076	geom	geo	750
0.1217	0.3443	1.2836	0.6595	0.5541	5.1932	geom	vect	750
0.4072	0.3103	2.4342	0.8656	0.588	7.1332	geom	vect-geo	800
0.4064	0.1129	2.4253	0.8749	0.3768	7.1282	geom	geo	800
0.408	0.6855	2.443	0.8562	1.0285	7.1382	geom	vect	800
0.3528	0.0839	2.0698	0.7442	0.454	6.789	geom	vect-geo	850
0.3529	0.3798	2.0796	0.75	0.5789	6.7953	geom	geo	850
0.3527	0.3957	2.0601	0.7384	0.7736	6.7827	geom	vect	850
0.6637	0.223	6.7509	1.262	0.77	13.0747	geom	vect-geo	900
0.6634	0.6604	6.7244	1.261	1.2462	13.0456	geom	geo	900
0.664	0.307	6.7774	1.263	0.6987	13.1037	geom	vect	900
0.4045	0.4423	9.7668	1.038	1.0076	19.7602	geom	vect-geo	950
0.4062	0.8606	9.741	1.0411	1.4293	19.7371	geom	geo	950
0.4032	0.2201	9.7927	1.0349	0.6976	19.7832	geom	vect	950
0.1354	0.5014	2.835	0.4075	0.7192	7.7986	geom	vect-geo	1000
0.1372	0.9583	2.8314	0.4254	1.1991	7.7824	geom	geo	1000
0.1336	0.0894	2.8386	0.3897	0.3051	7.8148	geom	vect	1000
0.3385	0.4122	2.9614	0.8135	0.7116	4.5045	plan(lin)	vect-geo	350
0.3384	0.4369	2.9616	0.8131	0.7471	4.5017	plan(lin)	geo	350
0.3386	0.3875	2.9613	0.8139	0.6761	4.5073	plan(lin)	vect	350
0.3672	0.4939	2.9175	0.8397	0.921	5.8305	plan(lin)	vect-geo	400
0.3671	0.4129	2.9178	0.8394	0.8408	5.8287	plan(lin)	geo	400
0.3672	0.575	2.9172	0.84	1.0013	5.8324	plan(lin)	vect	400
0.3257	0.1231	1.907	0.6747	0.3698	4.5866	plan(lin)	vect-geo	450
0.3257	0.0917	1.9078	0.6748	0.2529	4.5997	plan(lin)	geo	450
0.3257	0.2908	1.9062	0.6745	0.5218	4.5734	plan(lin)	vect	450
0.2926	0.1654	1.3191	0.5649	0.3405	2.9534	plan(lin)	vect-geo	500
0.2927	0.3656	1.3215	0.5656	0.5462	2.9718	plan(lin)	geo	500
0.2926	0.0615	1.3166	0.5642	0.195	2.9351	plan(lin)	vect	500
0.1559	0.4871	1.3956	0.4575	0.794	4.6192	plan(lin)	vect-geo	550
0.1559	0.6751	1.3942	0.459	1.0057	4.6265	plan(lin)	geo	550
0.1559	0.2994	1.3969	0.456	0.5824	4.6119	plan(lin)	vect	550
0.1637	0.4651	3.0381	0.6928	0.8456	6.1053	plan(lin)	vect-geo	600
0.164	0.6565	3.0365	0.6951	1.0506	6.1185	plan(lin)	geo	600
0.1634	0.2838	3.0396	0.6904	0.6407	6.092	plan(lin)	vect	600
0.1823	0.0958	3.002	0.6386	0.284	6.9057	plan(lin)	vect-geo	650
0.1824	0.1937	3.0038	0.6382	0.3873	6.9044	plan(lin)	geo	650
0.1823	0.1184	3.0003	0.639	0.402	6.907	plan(lin)	vect	650
0.4698	0.1894	3.9204	1.0315	0.6067	8.4806	plan(lin)	vect-geo	700
0.4698	0.2113	3.9169	1.0332	0.6416	8.4895	plan(lin)	geo	700
0.4698	0.318	3.9239	1.0298	0.8593	8.4717	plan(lin)	vect	700
0.4099	0.389	1.484	0.9804	0.6358	6.0695	plan(lin)	vect-geo	750
0.4099	0.2129	1.4836	0.9826	0.5459	6.0836	plan(lin)	geo	750
0.4099	0.5822	1.4843	0.9782	0.8538	6.0555	plan(lin)	vect	750
0.7105	0.5437	2.9222	1.1867	1.0627	7.9145	plan(lin)	vect-geo	800
0.7105	0.1967	2.9181	1.1886	0.609	7.9204	plan(lin)	geo	800
0.7105	0.9386	2.9262	1.1848	1.5164	7.9086	plan(lin)	vect	800
0.6644	0.2627	1.8776	1.0718	0.507	6.128	plan(lin)	vect-geo	850
0.6644	0.1508	1.8833	1.073	0.3914	6.1252	plan(lin)	geo	850
0.6644	0.6667	1.872	1.0706	0.9366	6.1308	plan(lin)	vect	850
0.9867	0.25	7.3324	1.598	0.7162	13.6609	plan(lin)	vect-geo	900

0.9866	0.4302	7.3201	1.5998	1.2124	13.6475	plan(lin)	geo	900
0.9867	0.5831	7.3447	1.5961	1.2119	13.6743	plan(lin)	vect	900
0.7158	0.3306	10.2402	1.3889	0.9634	20.2367	plan(lin)	vect-geo	950
0.7158	0.614	10.228	1.3912	1.3922	20.2299	plan(lin)	geo	950
0.7159	0.4041	10.2525	1.3867	1.2327	20.2435	plan(lin)	vect	950
0.2657	0.2153	2.925	0.67	0.5097	8.1233	plan(lin)	vect-geo	1000
0.2657	0.6702	2.9261	0.673	0.9962	8.1228	plan(lin)	geo	1000
0.2657	0.2775	2.9238	0.667	0.8593	8.1237	plan(lin)	vect	1000

(1) plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).

(2) vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.

(3) All data is given in millimetres.

Table A6. Data set 2 of errors (mm) obtained using different camera calibration methods and different extraction techniques and only *x*-axis data (f_x and f_y refined for planes methods).

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.1721	0.1911	4.2349	0.5463	0.8859	9.1321	plan(f)	vect-geo	350
0.1742	0.1963	4.225	0.5685	0.8527	9.1158	plan(f)	geo	350
0.1735	0.2067	4.2447	0.5242	0.9192	9.1483	plan(f)	vect	350
0.1301	0.2171	4.4856	0.5049	0.5416	9.6846	plan(f)	vect-geo	400
0.1293	0.2154	4.5001	0.4825	0.5953	9.7032	plan(f)	geo	400
0.1342	0.2216	4.471	0.5273	0.5734	9.6659	plan(f)	vect	400
0.1098	0.5133	2.2733	0.3006	1.0585	7.5968	plan(f)	vect-geo	450
0.1118	0.6996	2.3136	0.3152	1.2812	7.6719	plan(f)	geo	450
0.113	0.327	2.2352	0.3128	0.8357	7.5216	plan(f)	vect	450
0.1647	0.6899	1.7994	0.4659	0.9548	6.3631	plan(f)	vect-geo	500
0.1734	0.9182	1.8304	0.4684	1.2305	6.4797	plan(f)	geo	500
0.1582	0.4616	1.7715	0.4635	0.6821	6.2465	plan(f)	vect	500
0.2575	0.9358	2.5193	0.5667	1.3146	6.4115	plan(f)	vect-geo	550
0.2622	1.1709	2.4815	0.5216	1.5524	6.3047	plan(f)	geo	550
0.2554	0.7007	2.5582	0.6118	1.0768	6.5182	plan(f)	vect	550
0.2847	0.7972	3.9053	0.9314	1.5195	11.6181	plan(f)	vect-geo	600
0.2824	1.024	3.8297	0.9275	1.7418	11.499	plan(f)	geo	600
0.2893	0.5703	3.9841	0.9353	1.2973	11.7373	plan(f)	vect	600
0.2773	0.4108	2.4145	0.7625	0.7267	7.7977	plan(f)	vect-geo	650
0.2841	0.6074	2.3948	0.7605	0.9118	7.7039	plan(f)	geo	650
0.2705	0.2147	2.4362	0.7645	0.5416	7.8915	plan(f)	vect	650
0.2351	0.213	4.0315	1.1744	0.8602	11.7802	plan(f)	vect-geo	700
0.2289	0.4004	3.9563	1.1759	1.0942	11.645	plan(f)	geo	700
0.2423	0.1959	4.1076	1.1728	0.6695	11.9155	plan(f)	vect	700
0.2106	0.1641	3.6367	0.7539	0.5403	8.1608	plan(f)	vect-geo	750
0.2028	0.1924	3.5485	0.7528	0.6904	8.0091	plan(f)	geo	750
0.222	0.3612	3.7294	0.7551	0.8104	8.3125	plan(f)	vect	750
0.221	0.3988	2.9469	0.7897	0.7306	9.3193	plan(f)	vect-geo	800
0.21	0.1632	2.915	0.7776	0.4919	9.4763	plan(f)	geo	800
0.2349	0.6717	2.9812	0.8064	0.98	9.1623	plan(f)	vect	800
0.2531	0.1217	3.9398	1.0614	0.5754	10.5424	plan(f)	vect-geo	850
0.2427	0.4253	4.0092	1.0412	1.0259	10.7503	plan(f)	geo	850
0.2666	0.3757	3.8793	1.0815	0.7093	10.3346	plan(f)	vect	850

0.6946	0.3352	9.1388	1.7199	1.4826	16.7186	plan(f)	vect-geo	900
0.6785	0.7915	8.7974	1.6831	2.0305	16.3649	plan(f)	geo	900
0.7108	0.3851	9.4801	1.7567	0.9815	17.0722	plan(f)	vect	900
0.3524	0.3664	7.1309	1.2553	1.0416	13.9729	plan(f)	vect-geo	950
0.339	0.813	6.8514	1.2331	1.4861	13.6696	plan(f)	geo	950
0.3668	0.2454	7.4104	1.2776	0.7029	14.2762	plan(f)	vect	950
0.2003	0.2782	4.2172	0.9299	0.748	11.4059	plan(f)	vect-geo	1000
0.1959	0.7554	4.0803	0.9099	1.2361	11.7287	plan(f)	geo	1000
0.2096	0.2333	4.3751	0.95	0.6179	11.0831	plan(f)	vect	1000
0.168	0.1896	4.2345	0.5388	0.8692	9.1231	plan(fo)	vect-geo	350
0.1694	0.1968	4.2249	0.5597	0.8366	9.1071	plan(fo)	geo	350
0.17	0.2002	4.2441	0.5179	0.9018	9.1391	plan(fo)	vect	350
0.1296	0.2252	4.4845	0.5129	0.5396	9.6739	plan(fo)	vect-geo	400
0.1284	0.2213	4.4989	0.4918	0.5783	9.6923	plan(fo)	geo	400
0.1342	0.2322	4.4701	0.534	0.5894	9.6555	plan(fo)	vect	400
0.1092	0.4977	2.2704	0.3074	1.0413	7.5873	plan(fo)	vect-geo	450
0.11	0.6854	2.3102	0.3039	1.2641	7.6612	plan(fo)	geo	450
0.1124	0.3099	2.2328	0.3193	0.8185	7.5134	plan(fo)	vect	450
0.1607	0.6742	1.7959	0.4721	0.9376	6.35	plan(fo)	vect-geo	500
0.1684	0.9039	1.8262	0.4739	1.2133	6.465	plan(fo)	geo	500
0.1552	0.4446	1.7683	0.4702	0.6651	6.2351	plan(fo)	vect	500
0.2535	0.9201	2.5174	0.5619	1.2997	6.4123	plan(fo)	vect-geo	550
0.2573	1.1565	2.4797	0.5196	1.5399	6.3075	plan(fo)	geo	550
0.2518	0.6838	2.5556	0.6042	1.0594	6.5171	plan(fo)	vect	550
0.2821	0.7815	3.906	0.9378	1.504	11.6175	plan(fo)	vect-geo	600
0.2796	1.0095	3.8315	0.9335	1.7279	11.5004	plan(fo)	geo	600
0.2863	0.5535	3.9837	0.9421	1.28	11.7345	plan(fo)	vect	600
0.2715	0.3951	2.4115	0.7689	0.7111	7.7963	plan(fo)	vect-geo	650
0.2782	0.5928	2.3922	0.7664	0.8978	7.7042	plan(fo)	geo	650
0.2649	0.1984	2.433	0.7713	0.5244	7.8883	plan(fo)	vect	650
0.2345	0.2031	4.0328	1.1806	0.8452	11.7817	plan(fo)	vect-geo	700
0.2282	0.386	3.9585	1.1814	1.0791	11.6486	plan(fo)	geo	700
0.2415	0.2025	4.1076	1.1798	0.6524	11.9147	plan(fo)	vect	700
0.2139	0.1719	3.6377	0.7604	0.5565	8.1626	plan(fo)	vect-geo	750
0.2058	0.1821	3.551	0.7587	0.6752	8.0133	plan(fo)	geo	750
0.2244	0.3762	3.7293	0.762	0.827	8.3119	plan(fo)	vect	750
0.2274	0.4141	2.9441	0.7966	0.7443	9.3047	plan(fo)	vect-geo	800
0.2156	0.1727	2.9128	0.7846	0.5029	9.4592	plan(fo)	geo	800
0.2406	0.6882	2.9784	0.8133	0.9965	9.1502	plan(fo)	vect	800
0.2592	0.1209	3.9352	1.0685	0.56	10.5291	plan(fo)	vect-geo	850
0.2486	0.4106	4.0032	1.0485	1.0118	10.7337	plan(fo)	geo	850
0.2723	0.3916	3.8758	1.0885	0.7247	10.3246	plan(fo)	vect	850
0.7022	0.3256	9.1489	1.7277	1.4678	16.7211	plan(fo)	vect-geo	900
0.6863	0.7765	8.8131	1.6917	2.0176	16.3732	plan(fo)	geo	900
0.7181	0.3941	9.4847	1.7637	0.9978	17.0689	plan(fo)	vect	900
0.357	0.353	7.1392	1.2626	1.0263	13.9792	plan(fo)	vect-geo	950
0.3433	0.7978	6.8643	1.2404	1.4719	13.6813	plan(fo)	geo	950
0.3713	0.252	7.4141	1.2847	0.7191	14.2772	plan(fo)	vect	950
0.2	0.2632	4.2192	0.9372	0.7327	11.3883	plan(fo)	vect-geo	1000
0.1951	0.7402	4.0843	0.9171	1.2219	11.7062	plan(fo)	geo	1000
0.2086	0.2448	4.375	0.9572	0.6336	11.0705	plan(fo)	vect	1000
0.2758	0.2137	4.8862	0.8992	0.8118	7.506	plan(squ)	vect-geo	350

0.2831	0.2267	4.8743	0.8899	0.8754	7.4918	plan(squ)	geo	350
0.2721	0.2188	4.8981	0.9085	0.7482	7.5202	plan(squ)	vect	350
0.2343	0.2645	5.0988	0.756	0.7958	7.6853	plan(squ)	vect-geo	400
0.2429	0.2632	5.1111	0.7766	0.779	7.683	plan(squ)	geo	400
0.2273	0.2681	5.0866	0.7354	0.8125	7.6876	plan(squ)	vect	400
0.1602	0.3998	2.4289	0.5131	0.793	5.1609	plan(squ)	vect-geo	450
0.1723	0.5791	2.4995	0.5594	1.0136	5.2218	plan(squ)	geo	450
0.1496	0.2311	2.3613	0.4669	0.5723	5.1	plan(squ)	vect	450
0.2309	0.5654	1.2774	0.498	0.8445	4.2845	plan(squ)	vect-geo	500
0.2421	0.7868	1.3446	0.5457	1.1235	4.3684	plan(squ)	geo	500
0.22	0.344	1.2168	0.4504	0.5654	4.2006	plan(squ)	vect	500
0.3408	0.8013	1.5454	0.613	1.0424	3.2132	plan(squ)	vect-geo	550
0.3525	1.0292	1.5017	0.6012	1.2949	3.0823	plan(squ)	geo	550
0.3291	0.5734	1.5927	0.6248	0.8093	3.3441	plan(squ)	vect	550
0.2994	0.6531	3.3738	0.619	1.1985	8.1911	plan(squ)	vect-geo	600
0.3057	0.8726	3.2795	0.642	1.424	8.0768	plan(squ)	geo	600
0.2932	0.4341	3.4708	0.5961	0.973	8.3054	plan(squ)	vect	600
0.4051	0.2557	1.0035	0.5798	0.4572	3.1643	plan(squ)	vect-geo	650
0.4145	0.4447	1.0047	0.6172	0.6736	3.2005	plan(squ)	geo	650
0.3958	0.082	1.0062	0.5423	0.2438	3.2522	plan(squ)	vect	650
0.1969	0.155	3.4929	0.7447	0.5597	9.5534	plan(squ)	vect-geo	700
0.1994	0.2531	3.4173	0.7456	0.8221	9.4229	plan(squ)	geo	700
0.1969	0.2476	3.576	0.7438	0.709	9.6839	plan(squ)	vect	700
0.1303	0.2784	3.119	0.3216	0.7891	8.2832	plan(squ)	vect-geo	750
0.1389	0.1351	2.9885	0.3042	0.5249	8.1131	plan(squ)	geo	750
0.126	0.5186	3.251	0.3539	1.0534	8.4533	plan(squ)	vect	750
0.1172	0.5816	2.0373	0.4604	0.8944	5.8547	plan(squ)	vect-geo	800
0.1239	0.3159	1.9717	0.4519	0.6284	5.706	plan(squ)	geo	800
0.1144	0.8473	2.1051	0.4689	1.1604	6.0034	plan(squ)	vect	800
0.1463	0.1867	3.2672	0.5891	0.6552	11.0941	plan(squ)	vect-geo	850
0.1489	0.2524	3.3237	0.5691	0.6747	11.2867	plan(squ)	geo	850
0.148	0.5569	3.2185	0.6091	0.9561	10.9016	plan(squ)	vect	850
0.4965	0.2558	9.7205	1.2918	1.0738	13.792	plan(squ)	vect-geo	900
0.4819	0.5908	9.3837	1.2555	1.6216	13.4509	plan(squ)	geo	900
0.5119	0.4975	10.0573	1.3281	1.2026	14.133	plan(squ)	vect	900
0.2223	0.2257	7.6895	0.7842	0.6855	13.5263	plan(squ)	vect-geo	950
0.2168	0.5949	7.3818	0.7618	1.1278	13.193	plan(squ)	geo	950
0.2304	0.3352	7.9972	0.8065	0.9476	13.8596	plan(squ)	vect	950
0.2289	0.123	4.1217	0.5031	0.3857	11.1626	plan(squ)	vect-geo	1000
0.2357	0.5269	3.9084	0.5449	0.8711	10.8192	plan(squ)	geo	1000
0.2224	0.4204	4.3371	0.494	0.7195	11.5059	plan(squ)	vect	1000
0.2326	0.1819	3.5118	0.7939	0.7381	6.1694	geom	vect-geo	350
0.2345	0.1965	3.5022	0.7867	0.7851	6.1572	geom	geo	350
0.2316	0.1829	3.5213	0.8011	0.6912	6.1816	geom	vect	350
0.2129	0.2421	3.6596	0.6305	0.7065	6.2822	geom	vect-geo	400
0.2143	0.2307	3.674	0.6457	0.6737	6.2823	geom	geo	400
0.2115	0.26	3.6452	0.6152	0.7393	6.282	geom	vect	400
0.0847	0.4172	1.2554	0.3703	0.7408	3.6951	geom	vect-geo	450
0.0916	0.6085	1.2921	0.4064	0.9676	3.758	geom	geo	450
0.0802	0.2298	1.223	0.3342	0.514	3.6322	geom	vect	450
0.0932	0.5981	1.067	0.3387	0.8728	2.7557	geom	vect-geo	500
0.1024	0.8307	1.0228	0.3764	1.1523	2.8413	geom	geo	500

0.0857	0.3656	1.1171	0.301	0.5933	2.6701	geom	vect	500
0.1843	0.8487	2.4231	0.4417	1.133	4.8624	geom	vect-geo	550
0.1916	1.0874	2.3406	0.4358	1.381	4.7309	geom	geo	550
0.1785	0.61	2.507	0.4475	0.885	4.994	geom	vect	550
0.1932	0.7155	4.986	0.7482	1.3369	9.8435	geom	vect-geo	600
0.1927	0.9453	4.8778	0.7322	1.5743	9.7282	geom	geo	600
0.1937	0.4874	5.0943	0.7641	1.0996	9.9588	geom	vect	600
0.2069	0.3351	1.9374	0.4474	0.5952	4.8759	geom	vect-geo	650
0.2154	0.5342	1.8445	0.4423	0.8207	4.7866	geom	geo	650
0.1994	0.1445	2.032	0.4525	0.3697	4.9652	geom	vect	650
0.1738	0.1943	4.861	0.9643	0.7219	11.345	geom	vect-geo	700
0.1715	0.3469	4.7543	0.9607	0.9926	11.2133	geom	geo	700
0.1778	0.2205	4.9702	0.9679	0.7065	11.4767	geom	vect	700
0.2219	0.1968	4.9321	0.5608	0.7565	10.1973	geom	vect-geo	750
0.2144	0.1675	4.7914	0.5337	0.599	10.0261	geom	geo	750
0.2303	0.4177	5.0727	0.5878	1.0284	10.3685	geom	vect	750
0.2726	0.4546	3.1958	0.7144	0.8374	7.7799	geom	vect-geo	800
0.2643	0.1942	3.0683	0.7048	0.5617	7.63	geom	geo	800
0.2823	0.729	3.325	0.7241	1.1131	7.9298	geom	vect	800
0.3234	0.1189	3.2565	0.8609	0.4808	9.0934	geom	vect-geo	850
0.3143	0.3779	3.207	0.8398	0.9025	9.2869	geom	geo	850
0.3338	0.4222	3.313	0.8821	0.7954	8.8998	geom	vect	850
0.7762	0.3372	11.7855	1.5781	1.3109	15.8474	geom	vect-geo	900
0.76	0.7479	11.45	1.5431	1.8703	15.5075	geom	geo	900
0.7924	0.3938	12.1209	1.6131	1.1312	16.1874	geom	vect	900
0.4109	0.3494	9.8222	1.0878	0.9326	15.6659	geom	vect-geo	950
0.3974	0.7728	9.5152	1.0641	1.3838	15.3343	geom	geo	950
0.4245	0.2568	10.1291	1.1116	0.8608	15.9975	geom	vect	950
0.1767	0.2487	5.8435	0.7769	0.6501	13.3709	geom	vect-geo	1000
0.1704	0.7213	5.555	0.7551	1.144	13.029	geom	geo	1000
0.185	0.2462	6.1507	0.7988	0.6095	13.7129	geom	vect	1000
0.2303	0.1354	2.2305	0.7687	0.4611	4.9048	plan(lin)	vect-geo	350
0.2341	0.1428	2.2259	0.7606	0.5015	4.896	plan(lin)	geo	350
0.2274	0.1432	2.235	0.7768	0.4503	4.9136	plan(lin)	vect	350
0.218	0.1522	2.3895	0.5992	0.4401	5.0203	plan(lin)	vect-geo	400
0.2205	0.1398	2.4098	0.6209	0.4005	5.0248	plan(lin)	geo	400
0.2162	0.1811	2.3693	0.5775	0.4797	5.0158	plan(lin)	vect	400
0.0912	0.4875	1.0668	0.3329	0.7273	2.9538	plan(lin)	vect-geo	450
0.098	0.69	1.0444	0.3801	0.9961	2.8501	plan(lin)	geo	450
0.0918	0.285	1.0961	0.2857	0.5344	3.0575	plan(lin)	vect	450
0.0739	0.6576	1.829	0.2943	0.9346	3.7892	plan(lin)	vect-geo	500
0.0869	0.9016	1.7415	0.3429	1.2245	3.6714	plan(lin)	geo	500
0.0664	0.4135	1.9166	0.2716	0.6862	3.907	plan(lin)	vect	500
0.1416	0.8972	3.5569	0.3813	1.2795	6.1038	plan(lin)	vect-geo	550
0.1474	1.1477	3.4424	0.3703	1.5348	5.9626	plan(lin)	geo	550
0.1384	0.6466	3.6713	0.3922	1.0242	6.245	plan(lin)	vect	550
0.1702	0.7542	6.2128	0.8081	1.5379	11.0633	plan(lin)	vect-geo	600
0.1699	0.9963	6.0955	0.7903	1.7856	10.9397	plan(lin)	geo	600
0.1739	0.5242	6.3302	0.8258	1.2903	11.1868	plan(lin)	vect	600
0.1434	0.3634	3.1107	0.5137	0.7474	6.0753	plan(lin)	vect-geo	650
0.1512	0.5753	3.0036	0.5104	0.9888	5.978	plan(lin)	geo	650
0.1356	0.186	3.2178	0.517	0.5334	6.1726	plan(lin)	vect	650

0.2243	0.2556	6.0105	1.039	0.8839	12.5467	plan(lin)	vect-geo	700
0.2201	0.3955	5.8828	1.0387	1.1682	12.4051	plan(lin)	geo	700
0.2302	0.2727	6.1393	1.0394	0.8868	12.6883	plan(lin)	vect	700
0.3052	0.2353	6.1247	0.6412	0.9092	11.4044	plan(lin)	vect-geo	750
0.2945	0.2302	5.9729	0.6089	0.7652	11.2204	plan(lin)	geo	750
0.3159	0.4293	6.2765	0.6744	1.1966	11.5883	plan(lin)	vect	750
0.3652	0.456	4.2309	0.8044	0.981	8.9551	plan(lin)	vect-geo	800
0.3543	0.2154	4.0642	0.7949	0.692	8.7941	plan(lin)	geo	800
0.3761	0.7443	4.3977	0.8138	1.2893	9.1161	plan(lin)	vect	800
0.4132	0.1345	3.5197	0.9548	0.593	8.3984	plan(lin)	vect-geo	850
0.4013	0.3773	3.4392	0.9336	1.0639	8.135	plan(lin)	geo	850
0.425	0.4538	3.6046	0.9761	0.8524	8.732	plan(lin)	vect	850
0.8818	0.3865	12.9474	1.6777	1.4696	17.0166	plan(lin)	vect-geo	900
0.865	0.7519	12.5969	1.6403	2.0383	16.6615	plan(lin)	geo	900
0.8986	0.458	13.298	1.7151	1.3418	17.3717	plan(lin)	vect	900
0.5202	0.3745	10.9717	1.1951	1.0608	16.8097	plan(lin)	vect-geo	950
0.5061	0.7557	10.6494	1.1731	1.5245	16.4622	plan(lin)	geo	950
0.5344	0.334	11.294	1.2187	1.0728	17.1572	plan(lin)	vect	950
0.2705	0.2519	6.9299	0.8915	0.7674	14.5083	plan(lin)	vect-geo	1000
0.2605	0.6907	6.5932	0.87	1.2741	14.1497	plan(lin)	geo	1000
0.2829	0.3183	7.2665	0.9129	0.812	14.8669	plan(lin)	vect	1000

(1) plan(fo)= Modified Planes (150x150), plan(f)= Modified Planes (100x100), geo=geometric (Tsai), plan(lin)= Planes (linear), plan(squ)= Planes (quadratic).

(2) vect=vectorial method, geo= geometric method, vect-geo= vectorial geometric method.

(3) All data is given in millimetres.

APPENDIX B

Following, pictures corresponding to different tests performed are displayed.

TEST 1

Figure B-1. Area selected by user (test 1).

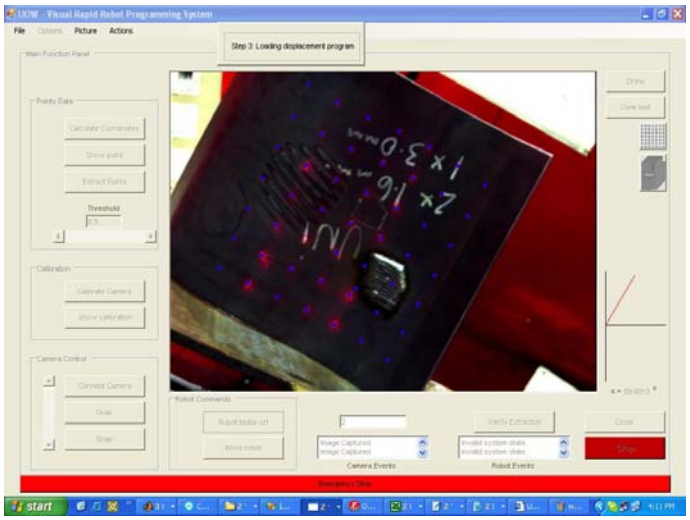


Figure B-2. First set of dots extracted (test 1).

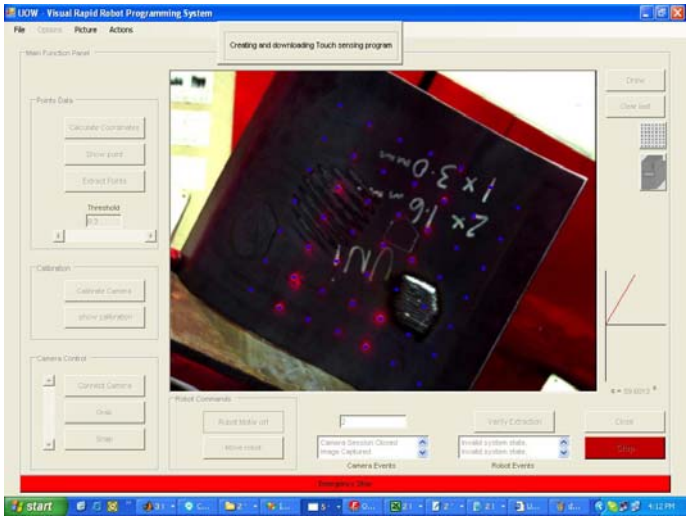


Figure B-3. Second set of dots extracted (test 1).

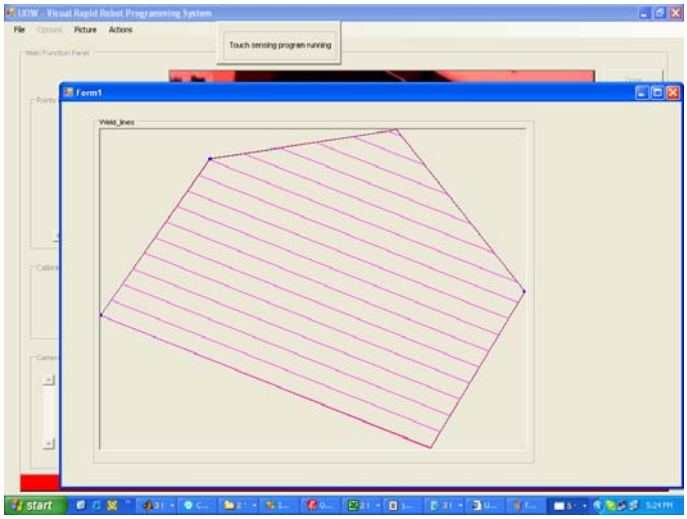


Figure B-3. 2-D plot of area and weld lines (test 1).

Figure B-4. 2-D Final welding result (test 1).

TEST 2

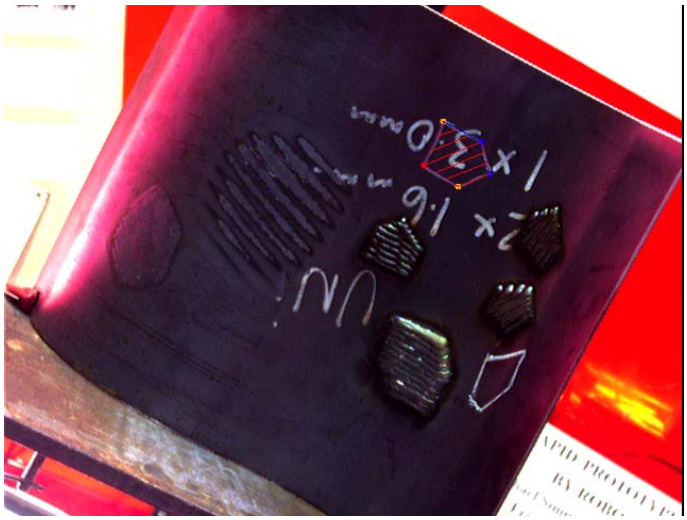


Figure B-5. Area selected by user (test 2).

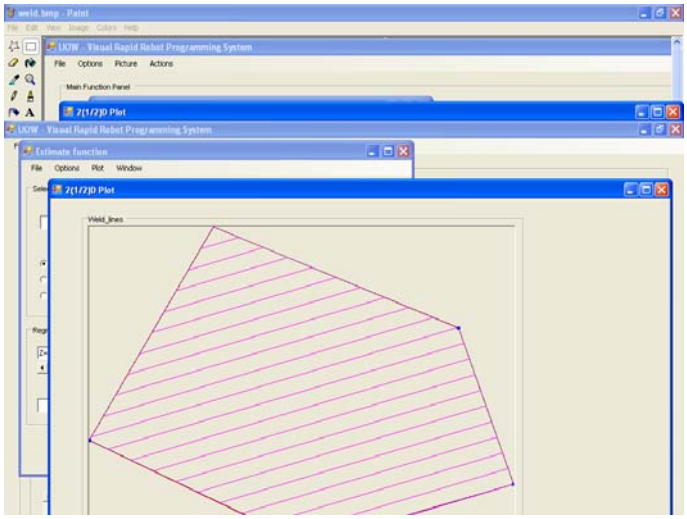


Figure B-6. 2-D plot of area and weld lines (test 2).

Figure B-7. 2-D Final welding result. (test 2).

APPENDIX C

The 3D coordinates estimation errors from 3 sets of data for the evaluation of the impact of distortion terms inclusion are presented in the following tables:

Table C1. Data set 1 of errors (mm) obtained with and without using distortion terms.

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method[(1)]	Error type[(2)]	Dist.
0.5628	0.6053	6.257	1.2194	1.4143	8.881	geo-dist	vect	400
0.4362	0.6515	5.0273	0.9689	1.3627	7.6158	geo-dist	geo	400
0.522	0.6586	10.9927	1.4846	1.8807	13.2827	geo-dist	vect	450
0.4866	0.6571	10.5553	1.341	1.8687	13.1305	geo-dist	geo	450
0.3048	0.5356	8.0768	0.9033	1.3425	11.4725	geo-dist	vect	500
0.2514	0.5522	7.0862	0.738	1.2704	10.4443	geo-dist	geo	500
0.378	0.4429	10.9378	1.1	1.1382	13.4294	geo-dist	vect	550
0.3343	0.4223	10.0365	0.9128	1.1016	12.851	geo-dist	geo	550
0.2618	0.2608	7.5893	0.7224	0.7353	12.6972	geo-dist	vect	600
0.1938	0.2259	6.284	0.5442	0.6596	11.3564	geo-dist	geo	600
0.3869	0.3409	10.257	1.3197	1.1705	15.178	geo-dist	vect	650
0.3098	0.3028	8.9363	1.147	1.1164	13.8588	geo-dist	geo	650
0.6452	0.3962	12.8159	1.5667	1.1542	19.3356	geo-dist	vect	700
0.6234	0.3859	12.4217	1.5214	1.1041	18.9889	geo-dist	geo	700
0.6351	0.2759	10.3367	1.5253	0.7832	13.6583	geo-dist	vect	750
0.6463	0.285	10.4605	1.5435	0.8118	14.0105	geo-dist	geo	750
0.8955	0.4356	16.0516	1.9712	1.1475	22.4069	geo-dist	vect	800
0.9137	0.4395	16.3642	2.0595	1.2229	22.8383	geo-dist	geo	800
0.7123	0.5588	20.7753	1.9161	1.4168	27.0132	geo-dist	vect	850
0.7932	0.5978	22.4095	2.0071	1.4525	27.827	geo-dist	geo	850
0.8805	0.9366	24.5551	2.126	2.332	31.8501	geo-dist	vect	900
0.9602	0.8994	26.17	2.2418	2.3967	34.3196	geo-dist	geo	900
0.5128	0.5934	5.5281	1.1674	1.4008	8.1515	geo	vect	400
0.3877	0.6379	4.2981	0.9169	1.3492	6.8864	geo	geo	400
0.4983	0.6495	10.2859	1.4342	1.8676	12.575	geo	vect	450
0.4646	0.648	9.8485	1.2906	1.8556	12.4228	geo	geo	450
0.2911	0.5249	7.3889	0.8543	1.3297	10.7842	geo	vect	500
0.2484	0.541	6.3981	0.689	1.2576	9.7554	geo	geo	500
0.3711	0.4386	10.2713	1.0525	1.1258	12.7621	geo	vect	550
0.3326	0.4168	9.3697	0.8653	1.0892	12.1847	geo	geo	550
0.2443	0.2603	6.9415	0.6763	0.7232	12.0497	geo	vect	600
0.1872	0.2236	5.6359	0.5018	0.6476	10.7084	geo	geo	600
0.3618	0.3394	9.6305	1.2751	1.1589	14.5516	geo	vect	650
0.292	0.3009	8.3095	1.1024	1.1048	13.2318	geo	geo	650
0.6121	0.3944	12.2105	1.5236	1.143	18.7311	geo	vect	700
0.5903	0.3839	11.8164	1.4784	1.0929	18.3845	geo	geo	700
0.5933	0.2766	9.7504	1.4836	0.7724	13.0727	geo	vect	750
0.6045	0.286	9.8746	1.5017	0.8224	13.4253	geo	geo	750
0.8583	0.4355	15.4878	1.931	1.137	21.8439	geo	vect	800
0.876	0.4399	15.8008	2.0194	1.2332	22.2761	geo	geo	800
0.6881	0.5588	20.2336	1.8776	1.4068	26.4721	geo	vect	850

0.7661	0.5981	21.8687	1.9686	1.4607	27.2871	geo	geo	850
0.8539	0.9291	24.035	2.089	2.3223	31.3311	geo	vect	900
0.9332	0.8924	25.6509	2.2047	2.3871	33.8023	geo	geo	900

(1) geo=geometric (Tsai), geo-dist=geometric (Tsai) with distortion factors.
(2)vect=vectorial method, geo= geometric method.
(3)All data is given in millimetres.

Table C2. Data set 2 of errors (mm) obtained with and without using distortion terms.

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.2164	0.2756	3.554	0.5574	0.6691	5.3265	geo-dist	vect	350
0.2785	0.3156	2.9969	0.5848	0.6724	4.9648	geo-dist	geo	350
0.2859	0.5111	7.1656	0.6773	1.3901	10.0164	geo-dist	vect	400
0.3402	0.5141	8.0988	0.814	1.4734	10.9759	geo-dist	geo	400
0.428	0.4277	9.7941	1.3291	1.2091	11.9713	geo-dist	vect	450
0.7228	0.5199	12.5417	1.8215	1.4136	14.6154	geo-dist	geo	450
0.4745	0.3952	10.2225	1.2649	0.9328	13.0102	geo-dist	vect	500
0.8644	0.5436	13.7912	1.9431	1.403	16.9584	geo-dist	geo	500
0.5173	0.446	8.1814	1.2999	1.1334	10.3457	geo-dist	vect	550
0.9203	0.6021	11.6917	1.7617	1.5928	14.5695	geo-dist	geo	550
0.5215	0.4133	6.7547	0.9771	1.0118	10.9621	geo-dist	vect	600
0.9435	0.57	10.573	1.5451	1.518	14.8819	geo-dist	geo	600
0.7214	0.3536	10.7172	1.5197	1.0047	13.4363	geo-dist	vect	650
1.0212	0.4491	13.5676	2.0085	1.3569	17.2207	geo-dist	geo	650
0.3409	0.2959	8.4057	0.8705	0.9141	12.2443	geo-dist	vect	700
0.8169	0.4215	13.7141	1.8029	1.1767	17.639	geo-dist	geo	700
0.3821	0.4713	10.4777	0.9598	1.113	12.9291	geo-dist	vect	750
0.8036	0.5132	15.4205	1.617	1.2857	18.522	geo-dist	geo	750
0.6342	0.9166	20.7124	1.789	2.1703	25.2765	geo-dist	vect	800
1.5231	1.0134	33.2042	3.3965	2.7437	38.7123	geo-dist	geo	800
0.7344	0.7873	24.9865	1.9202	2.1499	30.1894	geo-dist	vect	850
1.6508	1.0267	38.5556	3.7039	2.6902	45.0562	geo-dist	geo	850
0.6394	0.7161	23.8096	1.7646	1.8787	30.0761	geo-dist	vect	900
1.8133	1.062	42.3804	4.173	2.5707	49.5281	geo-dist	geo	900
0.6787	0.4959	19.6841	1.6082	1.5238	28.1221	geo-dist	vect	950
1.8533	0.8924	37.0336	3.664	2.1709	44.677	geo-dist	geo	950
1.1816	0.7329	31.1391	2.6311	2.1531	38.4253	geo-dist	vect	1000
2.4419	1.1716	50.4551	5.0699	3.0225	58.7091	geo-dist	geo	1000
0.2263	0.2714	3.2702	0.573	0.6639	5.0439	geo	vect	350
0.2923	0.3104	2.7129	0.6004	0.6672	4.6823	geo	geo	350
0.2851	0.5072	6.9	0.6919	1.3853	9.7524	geo	vect	400
0.3362	0.5106	7.8335	0.7996	1.4686	10.7124	geo	geo	400
0.4227	0.4263	9.5465	1.3157	1.2046	11.7218	geo	vect	450
0.7135	0.5196	12.2949	1.8081	1.409	14.3702	geo	geo	450
0.4671	0.3953	9.992	1.2524	0.9371	12.7781	geo	vect	500
0.8541	0.5444	13.5619	1.9306	1.4072	16.7279	geo	geo	500
0.5072	0.4489	7.9672	1.2883	1.1374	10.131	geo	vect	550
0.909	0.6049	11.4787	1.7501	1.5967	14.3566	geo	geo	550
0.511	0.4163	6.557	0.9664	1.0154	10.7664	geo	vect	600
0.9327	0.5726	10.3766	1.5344	1.5216	14.6879	geo	geo	600
0.7122	0.3539	10.5379	1.51	1.008	13.2564	geo	vect	650
1.0114	0.4496	13.3892	1.9988	1.3601	17.0422	geo	geo	650
0.335	0.2948	8.2426	0.8617	0.9111	12.0803	geo	vect	700

0.8086	0.4214	13.5528	1.7942	1.1737	17.4772	geo	geo	700
0.3769	0.4693	10.3323	0.952	1.1103	12.7854	geo	vect	750
0.7967	0.5124	15.2768	1.6092	1.283	18.3783	geo	geo	750
0.631	0.9146	20.5874	1.7823	2.168	25.1513	geo	vect	800
1.5178	1.0127	33.0835	3.39	2.7415	38.5918	geo	geo	800
0.7317	0.7863	24.8799	1.9145	2.1479	30.0837	geo	vect	850
1.6463	1.0263	38.4538	3.6984	2.6883	44.9559	geo	geo	850
0.6376	0.7153	23.7197	1.7598	1.877	29.9875	geo	vect	900
1.8096	1.0619	42.2969	4.1685	2.5692	49.4466	geo	geo	900
0.6759	0.4956	19.6097	1.6042	1.5225	28.0516	geo	vect	950
1.8498	0.8924	36.9653	3.6603	2.1721	44.6124	geo	geo	950
1.1793	0.7329	31.0857	2.6283	2.1522	38.3726	geo	vect	1000
2.4395	1.1717	50.4084	5.0674	3.0217	58.6662	geo	geo	1000

(1) geo=geometric (Tsai), geo-dist=geometric (Tsai) with distortion factors.
(2)vect=vectorial method, geo= geometric method.
(3)All data is given in millimetres.

Table C3. Data set 3 of errors (mm) obtained with and without using distortion terms.

Average *x*	Average *y*	Average *z*	Maximum *x*	Maximum *y*	Maximum *z*	Cal. Method(1)	Error type(2)	Dist.
0.1634	0.1428	3.791	0.6482	0.6305	6.7828	0.1634	0.1428	350
0.1525	0.144	3.401	0.6029	0.6713	6.5136	0.1525	0.144	350
0.2892	0.3337	6.5271	0.8712	1.0527	8.9144	0.2892	0.3337	400
0.3594	0.3656	7.2536	1.0515	1.181	9.9792	0.3594	0.3656	400
0.6022	0.4031	10.2848	1.3028	1.1086	12.7368	0.6022	0.4031	450
1.1138	0.5899	14.459	2.1871	1.6335	17.1191	1.1138	0.5899	450
0.7275	0.4364	11.1585	1.5983	1.1734	14.2854	0.7275	0.4364	500
1.3809	0.674	16.6934	2.7147	1.9101	19.8568	1.3809	0.674	500
0.7972	0.5215	10.1102	1.6043	1.2884	13.588	0.7972	0.5215	550
1.4761	0.749	16.1836	2.6012	1.984	19.9608	1.4761	0.749	550
0.672	0.3804	7.6162	1.2922	1.0564	10.8767	0.672	0.3804	600
1.3355	0.5966	13.7599	2.276	1.7476	16.8269	1.3355	0.5966	600
0.7371	0.3021	9.794	1.3949	0.7107	12.043	0.7371	0.3021	650
1.3042	0.4842	15.3241	2.3511	1.3203	17.2367	1.3042	0.4842	650
0.4782	0.3128	8.8098	1.1592	0.7699	12.9544	0.4782	0.3128	700
1.1034	0.4392	15.5476	2.1843	1.062	20.1514	1.1034	0.4392	700
0.479	0.592	10.7362	1.248	1.4875	15.7596	0.479	0.592	750
1.1571	0.63	18.7787	2.7643	2.0155	25.4371	1.1571	0.63	750
0.5292	0.9073	14.5553	1.6909	1.9735	19.1073	0.5292	0.9073	800
1.2304	0.8756	23.8273	2.9758	2.4107	28.8733	1.2304	0.8756	800
0.8462	0.8266	25.262	2.6401	2.3905	35.2368	0.8462	0.8266	850
1.926	1.0747	40.7485	4.9296	3.1125	52.3593	1.926	1.0747	850
0.6899	0.7535	23.3939	2.1533	1.953	29.6037	0.6899	0.7535	900
2.2348	1.1535	47.2005	5.1164	3.2734	58.0332	2.2348	1.1535	900
0.8116	0.6089	22.4631	2.2112	1.8748	31.1211	0.8116	0.6089	950
2.178	1.0032	43.0808	4.7737	2.7395	54.5968	2.178	1.0032	950
1.0797	0.7592	28.8834	2.5161	2.044	35.6878	1.0797	0.7592	1000
2.5137	1.1615	51.3334	5.3523	2.9061	63.672	2.5137	1.1615	1000
0.1594	0.1464	2.5129	0.5984	0.6072	5.5109	0.1594	0.1464	350
0.1575	0.1403	2.1275	0.5529	0.6481	5.242	0.1575	0.1403	350
0.2785	0.3192	5.3268	0.8245	1.0308	7.7195	0.2785	0.3192	400
0.3379	0.3519	6.0541	1.0045	1.1592	8.7865	0.3379	0.3519	400
0.572	0.4083	9.1638	1.259	1.1294	11.6178	0.572	0.4083	450

1.074	0.5957	13.344	2.1433	1.654	16.0074	1.074	0.5957	450
0.6939	0.4431	10.1125	1.5574	1.1927	13.2472	0.6939	0.4431	500
1.3412	0.6807	15.6554	2.6741	1.929	18.827	1.3412	0.6807	500
0.7611	0.5346	9.1364	1.5663	1.3064	12.6204	0.7611	0.5346	550
1.438	0.7603	15.2185	2.5634	2.0017	19.0016	1.438	0.7603	550
0.6363	0.3936	6.7125	1.257	1.073	9.9816	0.6363	0.3936	600
1.3001	0.6065	12.8651	2.2408	1.7639	15.9333	1.3001	0.6065	600
0.7063	0.3028	8.9673	1.3626	0.7259	11.2216	0.7063	0.3028	650
1.2718	0.4866	14.5054	2.319	1.3352	16.4224	1.2718	0.4866	650
0.4547	0.3041	8.0554	1.1299	0.756	12.2088	0.4547	0.3041	700
1.074	0.4361	14.8029	2.1551	1.0483	19.4168	1.074	0.4361	700
0.4599	0.5806	10.0584	1.2217	1.4752	15.0924	0.4599	0.5806	750
1.1324	0.6249	18.1126	2.7385	2.0035	24.785	1.1324	0.6249	750
0.5144	0.8965	13.9569	1.6678	1.9627	18.5156	0.5144	0.8965	800
1.2101	0.8687	23.2424	2.953	2.4001	28.2949	1.2101	0.8687	800
0.8348	0.8212	24.7532	2.6206	2.3813	34.7466	0.8348	0.8212	850
1.9098	1.073	40.2624	4.9108	3.1037	51.8937	1.9098	1.073	850
0.6807	0.7491	22.956	2.1365	1.945	29.1764	0.6807	0.7491	900
2.2208	1.1523	46.7977	5.1011	3.2662	57.648	2.2208	1.1523	900
0.8017	0.6061	22.0976	2.1973	1.8682	30.7638	0.8017	0.6061	950
2.1656	1.003	42.7456	4.7607	2.7335	54.2743	2.1656	1.003	950
1.0714	0.757	28.6011	2.5051	2.0389	35.4112	1.0714	0.757	1000
2.5045	1.161	51.0842	5.343	2.9017	63.4386	2.5045	1.161	1000

(1) geo=geometric (Tsai), geo-dist=geometric (Tsai) with distortion factors.

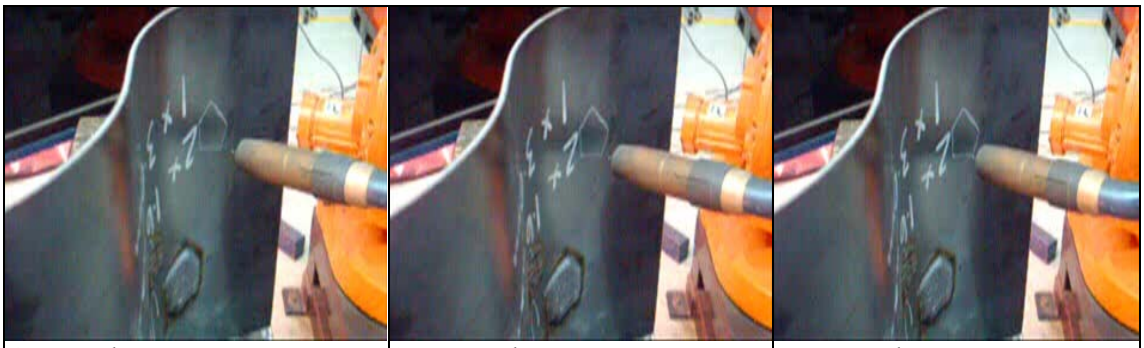
(2)vect=vectorial method, geo= geometric method.

(3)All data is given in millimetres.

APPENDIX D

A sequence of shots showing the approach of the tool to the work piece is shown below. The speed is reduced when the torch reach a distance where it is likely for the work piece to be located. Time equal to zero is the moment when the sequence of shots starts and does not represent any particular event. At time equal 17.53 seconds, the wire makes contact with the work piece and the coordinate of the point is stored in the robot controller to be afterwards read by the system's software. After the surface is detected, a backwards motion is performed in order to repeat the process for the next point

		
0 seconds	0.6 seconds	1.2 seconds
		
2 seconds	3 seconds	4 seconds



6 seconds	10 seconds	15 seconds

17.53 seconds	18.72 seconds

Figure D-1. Sequence of shots of a tactile (Touch) sensing process.

APPENDIX E: EQUIPMENT SPECIFICATIONS

POWER SUPPLY

The technical specifications of the Fronius "TransPuls Synergic 5000" are provided in the table below.

Table E1. Welding Power supply technical data. [84]

Supply Voltage		3x200-400V 3x380-460V
Supply Voltage Tolerance		±10%
Supply Voltage Frequency		50/60 Hz
Fuse Protection		65/35 A Slow blow
Primary continuous current (100% D.C.)		10.1-36.1 A
Primary continuous power		12.4-13.9kVa
Cos phi		0.99
Efficiency		88-91%
Welding Current Range	MIG/MAG Rod Electrode (MMA) TIG	3-500 A 10-500 A 3-500 A
Welding Current at 10min/40°C (104°F)	MIG/MAG Rod Electrode (MMA) TIG	500 A 450 A 320-340 A
Welding Voltage	MIG/MAG Rod Electrode (MMA) TIG	14.2-39 V 20.4-40 V 10.1-30 V
Maximum Welding Voltage		-
Open-Circuit Voltage		68-78 V
Degree of protection		IP23
Type of Cooling		AF
Insulation Class		F
Marks of Conformity		CE,CSA
Safety designation		S
Measurements (Length, width, height)		625 x 290 x 475 mm
Weight		35.6 Kg

ROBOTIC ARM

The technical specifications of the ABB “IRB1400” Robotic arm used are given in the table below.

Table E1. Robot technical data. [85]

Supply Voltage		200-600V, 50/60Hz
Handling capacity		5 Kg
Reach		1.44 m
Supplementary Load	On axis 3 On axis 1	18 Kg 19 Kg
Number of axes	Manipulator External	6 6
Position Repeatability		0.05 mm
Maximum TCP Velocity		2.1 m/s
Mounting		Floor
Robot base dimensions		620 x 450 mm
Weight		225 kg

APPENDIX F. OPTICS

The optical elements used play an important role in the development of any vision system; therefore it is important to give a brief review of some important concepts related to optics

LENSES

A lens is a refracting device made of a transparent material such as glass. There are two basic kinds of lenses are Concave and Convex. Concave lenses are thinner at the centre and its thickness increases towards the outer edges, these lenses are also called diverging or negative. A convex lens, on the other hand, is thicker at the centre and the thickness decreases towards the edges, they are also called converging or positive lenses [41]. Figure F.1 illustrates the more common types of lenses.

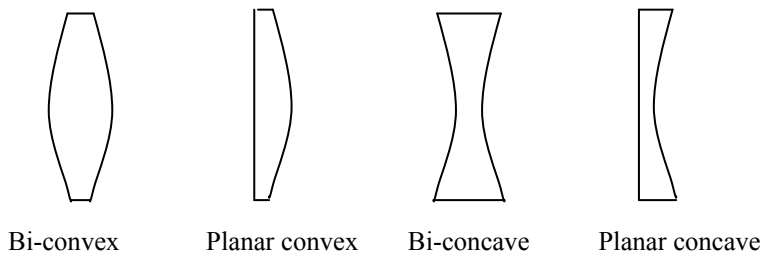


Figure F.1. Convex and Concave Lenses.

When incident parallel rays of light pass through a lens, they converge into a point due to refraction as illustrated in Figure F.2. The distance from this point to the lens is called the *focal length.* In other words, it is the distance at which an image for an object located at the infinity is formed [42].

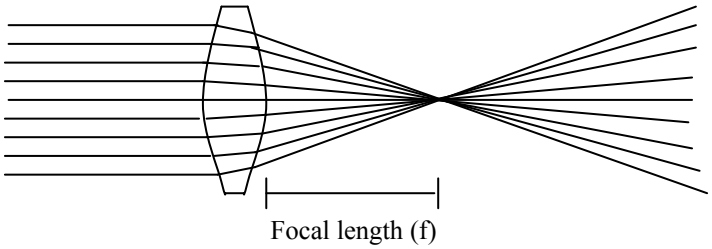


Figure F.2. Focal length of a lens.

DIFFRACTION

Frank and Leno Pedrotti define diffraction as [42]: *"Any deviation from geometrical optics that results from the obstruction of a wavefront of light"*. The Huygens-Fresnel principle affirms: *every unobstructed point of a wavefront of light can be a source of spherical secondary wavelets"*. From this principle it can be deduced that the diffraction effect is better observed when the size of the aperture through which the light passes the obstructing object is smaller than the wavelength of the incident light. This means that the smaller the aperture, the more likely the diffracted waves will approach a circular wavefront [41]. Diffraction can occur even in the presence of non opaque objects which can cause variations in the amplitude or phase of the light [42].

Using the diffraction phenomena, it is possible to manipulate light to produce different patterns. If it is desired to create a periodical pattern, then an assembly of repetitive diffracting elements can be used. Such an assembly is known as a *"Diffraction grating"*. In a simplified view, the effect of using a diffraction grating is to generate a set of points of maximum value (*principal maxima*) from the interaction of the different circular wavefronts; while at other points these wavefronts eliminate each other. This effect is illustrated in Figure F.3.

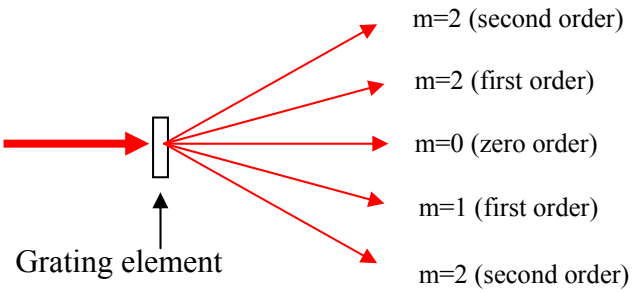


Figure F.3. Effect produced by a diffraction grating element . [86]

The following equation governs the relationship between the order of the principal maxima (m) and the diffraction angle:

$$a[\sin(\theta_i) + \sin(\theta_m)] = m\lambda$$

[Equation 116]

In the equation above, θ_i is the incident angle, θ_m is the diffraction angle, a is the period of the grating and λ the wavelength of the incident light. If the incident light is perpendicular as in Figure 2.1.2-1, equation 116 becomes:

$$a\sin(\theta_m) = m\lambda$$

[Equation 117]

Using this equation it is possible to calculate the angle of each principal maximum. It should be noted that this varies according to the wavelength of the incident light. By increasing the number of slits in the grating (N) the principal maxima get sharper, by increasing the separation between slits the principal maxima are closer together and by increasing the size of the slits the number of visible maxima diminishes [42].

THE CAMERA

A camera is an optical system that allows capturing an image. The most basic type of camera is the *pinhole* camera. This camera basically consists of an enclosed box which allows the light to pass trough a small hole. In this way, the light entering the box forms an inverted image at the rear, where a film plate is located as shown in Figure F.4.

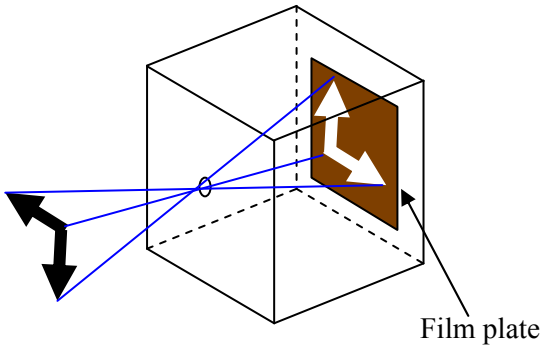


Figure F.4. Pinhole camera.

A pinhole camera can obtain images of objects in focus from an unlimited distance since it does not require any focusing. The *depth of field* of this kind of cameras is said to be unlimited. The depth of field is the range of distances between which all objects appear in focus in the image. The *aperture* is the opening that controls the amount of light entering into the system (i.e. the camera). If, as in a modern camera, a lens is used to focus the rays of light to form the image, a quantity known as *relative aperture* (A) is commonly used and is defined by the ratio between the focal length (f) of the lens and the diameter of the aperture (D).

$$A \equiv \frac{f}{D}$$

[Equation 118]

The relative aperture is also known as *focal ratio* or *f-number* and the depth of field is dependant of this value. Many commercial camera lenses have adjustable f-number by means of a diaphragm of variable diameter and the focus can be adjusted by changing the position of the lens.

APPENDIX G. LASER SAFETY

The word laser originates from the acronym LASER: **L**ight **A**mplification by **S**timulated **E**mission of **R**adiation [87]. The term laser is used to describe a device which produces optical radiation which is spatially and temporally coherent, quasimonochromatic (i.e. it has a very narrow bandwidth), has a large radiant power and is directional [41]. Laser radiation can be generated at diverse wavelengths.

The fact that a laser can produce a very high power density[88], makes it capable of producing injuries to the human body if exposed to this radiation. Therefore safety measures should be taken when operating these devices. Henderson [88] provides a table indicating the tissues at risk when exposed to laser radiation for different wavelength ranges (Table G.1).

Table G.1. Tissues affected by laser exposure. [88]

Waveband (nm)	Tissue		
	Skin	Outer Layers of Eye	Retina
<700	x	x	
700-1400	x	x	x
>1400	x	x	

Lasers can be classified in different categories depending on their potential to cause damage to human tissues. The *International and Electromechanical Commission* (**IEC**) developed a laser classification that is internationally accepted. This classification has six levels as follows [88-90]:

- Class 1: Lasers in this category are considered safe under any condition. These lasers have a very low output power, in the order of micro Watts, or are completely

enclosed such that there is not access to the laser radiation under normal operation conditions.

- Class 1M: These lasers have the same characteristics of Class1 lasers, except that they can become hazardous if an optical element is placed within the beam.
- Class 2: Lasers that fall in this category, produce visible light and low power (less than 1.0 mW). Direct exposure of the eye to this light does not cause damage if the expose time is less than .025 seconds which is enough time for a human to react by involuntary blinking (0.1 seconds).
- Class 2M: These lasers have the same characteristics of Class2 lasers, except that they can become hazardous if an optical element is placed within the beam.
- Class 3R: This category covers visible and invisible laser radiation. Direct viewing of the beam is potentially dangerous. These lasers have an accessible emission limit (AEL), i.e. the maximum allowed optical power, five times the AEL of Class2 for visible light and 5 times the AEL for Class1 for other wavelengths.
- Class 3B: Lasers on this category are hazardous if viewed directly without eye protection. Diffuse reflections are normally safe. These lasers have a maximum power of 0.5 W for wavelengths above 315nm.
- Class 4: These lasers produce hazardous diffuse reflections. They can produce skin injuries and are a fire risk. They must be used with extreme caution.

Depending on the laser classification, laser systems should be properly labelled and protected with interlocks and a “Key” control.

Protective eyewear must be used in areas where a Class 3B or Class 4 laser is in use. If a Class 3R laser emitting non-visible optical radiation is in operation, eyewear must also

be used [90]. Eyewear should be selected in a way that the incident light into the eye falls to safe levels but still enough light is perceived in order to achieve good vision. The Eyewear should be chosen according to the maximum power and wavelength of the laser. Thus, if a laser emits visible light, the eyewear should provide enough attenuation to decrease the optical power reaching the eye to 1mW or less, if the laser emits invisible radiation, this power should be 0.56mW or less [89]. In addition, protective eyewear is designed for specific wavelengths; therefore the attenuation level is not the same for wavelengths outside the value specified by the manufacturer. Table G.2 indicates the equivalence between optical density and attenuation factor for protective eyewear, which is in fact logarithmic. Optical density is a measure which indicates the amount of radiation that protective eyewear allows passing and is a logarithmic function of the inverse of the transmittance.

Table G.2. Equivalence Between attenuation factor an optical density. [89]

OD	Attenuation Factor
1	10
2	100
3	1000
4	10000
5	100000
6	1000000
7	10000000
8	100000000

APPENDIX H. SPATIAL ROTATION AND TRANSLATION

It is necessary to review the spatial translation and rotation topics as these are important issues in the three dimensional problem addressed in the current work.

Translation can be defined as the displacement of a point in space by a defined distance along a given direction vector [20]. Figure H.1 shows the translation of a point, relative to a reference frame, by a distance t along the unit vector $\hat{\mathbf{u}}$, the translation would be given by:

$$\mathbf{T} = t\hat{\mathbf{u}} = t\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} tu_x \\ tu_y \\ tu_z \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

[Equation 119]

A rotation can be considered as revolution of an object around a defined vector by a given number of degrees. A rotation can be expressed as a combination of individual rotations around the x, y and z axis of a reference coordinate frame as indicated in Equation 120. The rotation angle around x, y and z are represented by θ_1 θ_2 and θ_3 respectively. These rotations are known as Roll, Pitch and Yaw [20]. The Matrix **R** is known as rotation matrix.

$$\mathbf{R} = \begin{bmatrix} \cos(\theta_2)\cos(\theta_3) & \sin(\theta_1)\sin(\theta_2)\cos(\theta_3) - \cos(\theta_1)\sin(\theta_3) & \cos(\theta_1)\sin(\theta_2)\cos(\theta_3) + \sin(\theta_1)\sin(\theta_3) \\ \cos(\theta_2)\sin(\theta_3) & \sin(\theta_1)\sin(\theta_2)\sin(\theta_3) + \cos(\theta_1)\sin(\theta_3) & \cos(\theta_1)\sin(\theta_2)\sin(\theta_3) - \sin(\theta_1)\cos(\theta_3) \\ \sin(\theta_2) & \sin(\theta_1)\cos(\theta\ ) & \cos(\theta_1)\cos(\theta_2) \end{bmatrix}$$

[Equation 120]

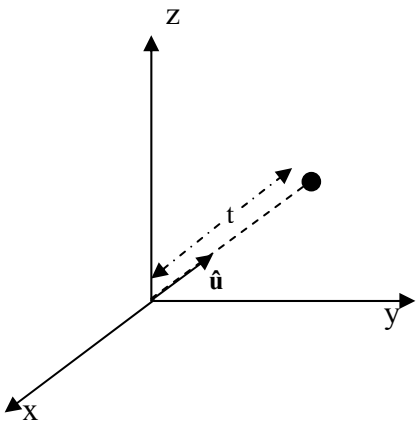


Figure H.1. Translation of a point.

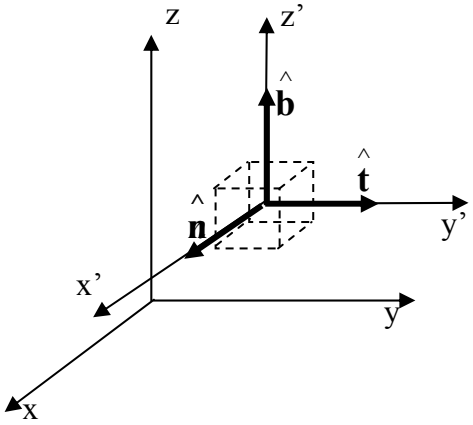


Figure H.2. Coordinate Frame Attached to an Object.

If a coordinate frame is defined and attached to an object, it is possible to express its position in the space in terms of rotation and translation relative to a reference coordinate frame or a World frame. If three unit vectors are considered, each one along one of the object's coordinate frame (x'y'z') axis (Figure H.2), the rotation matrix can be expressed as follows:

$$\mathbf{R} = \begin{bmatrix} \hat{\mathbf{n}} & \hat{\mathbf{t}} & \hat{\mathbf{b}} \end{bmatrix}$$

[Equation 121]

By using the rotation matrix **R**, any point **P'** with object coordinates (x', y', z'), can be expressed in terms of world coordinate frame coordinates by using the following relationship:

$$\mathbf{P} = \mathbf{RP'}+\mathbf{T}$$

[Equation 122]

A rotation can also be expressed in terms of *Quaternions*. A quaternion is a condensed form for describing a rotation and is defined as:

$$q = q_1 + \mathbf{u} = q_1 + (q_2\,\hat{\mathbf{i}} + q_3\,\hat{\mathbf{j}} + q_4\,\hat{\mathbf{k}})$$

[Equation 123]

$$q_1 = \cos\left(\alpha/2\right)$$

[Equation 124]

$$\mathbf{u} = \sin\left(\alpha/2\right)\hat{\mathbf{n}}$$

[Equation 125]

Where **n** is a unit vector around which the rotation is being made and α is the rotation angle. A quaternion that is compliant of Equations 124 and 125 is called unit quaternion since the sum of the squared value of its elements is one [91] .A rotation of a vector **P'** can be expressed as**:**

$$\mathbf{P} = q\mathbf{P}q'$$

[Equation 126]

$$q' = q_1 - \mathbf{u}$$

[Equation 127]

The product between two quaternions, q and p, is defined as follows:

$$qp = (q_1 p_1 - \mathbf{u}_q \bullet \mathbf{u}_p) + \left(q_1\mathbf{u}_p + p_1\mathbf{u}_p + \mathbf{u}_q \otimes \mathbf{u}_p\right)$$

[Equation 128]

A quaternion can be found from the rotation matrix **R** as follows [92]:

$$q1 = \frac{\sqrt{n_1 + t_2 + b_3 + 1}}{2}$$

[Equation 129]

$$q2 = \frac{\sqrt{n_1 - t_2 - b_3 + 1}}{2} \qquad sign(q_2) = sign(t_3 - b_2)$$

[Equation 130]

$$q3 = \frac{\sqrt{t_2 - n_1 - b_3 + 1}}{2} \qquad sign(q_3) = sign(b_1 - n_3)$$

[Equation 131]

$$q4 = \frac{\sqrt{b_3 - n_1 - t_2 + 1}}{2} \qquad sign(q_4) = sign(n_2 - t_1)$$

[Equation 132]

With $\hat{\mathbf{n}} = [n_1;\ n_{2;}\ n_3]$, $\hat{\mathbf{t}} = [t_1;\ t_{2;}\ t_3]$ and $\hat{\mathbf{b}} = [b_1;\ b_{2;}\ b_3]$ Similarly, the rotation matrix can be found the correspondent quaternions by the following relationship:

$$\mathbf{R} = \begin{bmatrix} q_1^2 + q_2^2 + q_3^2 + q_4^2 & 2(q_2q_3 - q_1q_4) & 2(q_1q_3 + q_2q_4) \\ 2(q_1q_4 + q_2q_3) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_1q_2 + q_3q_4) & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{bmatrix}$$

[Equation 133]

APPENDIX I. MINIMIZATION OF THE DISTANCE FROM A POINT TO A LINE

The distance *d* from a point **T** to a line **L** described by equation 134 can be found by using equation 135.

$$\mathbf{L} = \mathbf{P}_1 + \beta\hat{\mathbf{m}}$$

[Equation 134]

$$d = \sqrt{\|\mathbf{T} - \mathbf{P}_1\|^2 - [(\mathbf{T} - \mathbf{P}_1) \bullet \hat{\mathbf{m}}]^2}$$

[Equation 135]

Equation 135 can be more easily explained by referring to Figure I.1. According to such Figure it is possible to find the distance *d* by using the Pythagoras theorem:

$$d = \sqrt{\overline{\mathbf{TP}_1}^2 - \overline{\mathbf{TP}_2}^2}$$

[Equation 136]

The length of the hypotenuse ($\overline{\mathbf{TP}_1}$) and the cathetus $\overline{\mathbf{TP}_2}$ can be found by applying Equations 137 and 138

$$\overline{\mathbf{TP}_1} = \|\mathbf{T} - \mathbf{P}_1\|$$

[Equation 137]

$$\overline{\mathbf{TP}_2} = (\mathbf{T} - \mathbf{P}_1) \bullet \hat{\mathbf{m}}$$

[Equation 138]

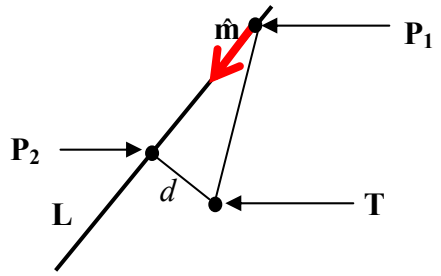


Figure I.1. Distance form a pint to a line.

If **T** = (t1, t2, t3), **P1**=(p1x,p2x,p3x) and **m̂** = (m1, m2, m3) then Equation 135 can be rewritten as follows:

$$d^2 = (t1 - p1x)^2 + (t2 - p1y)^2 + (t3 - p1z)^2 - [(t1 - p1x)m1 + (t2 - p1y)m2 + (t3 - p1z)m2]^2$$

[Equation 139]

By expanding the terms in Equation 139 it is obtained:

$$d^2 = (t1 - p1x)^2 + (t2 - p1y)^2 + (t3 - p1z)^2 - m1^2(t1 - p1x)^2 - 2m1m2(t1 - p1x)(t2 - p1y) - m2^2(t2 - p1y)^2 - 2m3(t3 - p1z)[m1(t1 - p1x) + m2(t2 - p1y)] - m3^2(t3 - p1z)^2$$

[Equation 140]

By further expanding terms in equation 141:

$$d^2 = t1^2 - 2t1p1x + p1x^2 + t2^2 - 2t2p1y + p1y^2 + t3^2 - 2t3p1z + p1z^2 - m1^2t1^2 + 2m1^2t1p1x - m1^2p1x^2 - 2m1m2t1t2 + 2m1m2t1p1y + 2m1m2p1xt2 - 2m1m2p1xp1y - m2^2t2^2 + 2m2^2t2p1y - b^2p1y^2 - 2m1m3t1t3 + 2m1m3t1p1z + 2m1m3t3p1x - 2m1m3p1xp1z - 2m2m3t2t3 + 2m2m3t2p1z + 2m2m3t3p1y - 2m2m3p1yp1z - m3^2t3^2 + 2m3^2t3p1z - m3^2p1z^2$$

[Equation 141]

To minimize the equation for the distance *d*, it is found the partial derivatives of Equation 141 with respect to *t*1, *t*2 and *t*1 and equated to zero since the aim of this process is to find an unknown point T which distance is minimal to several lines according to the two planes calibration method.

$$\frac{\delta d^2}{\delta t1} = 2t1(1-m1^2) - 2m1m2t2 - 2m1m3t3 + 2(-p1x^2 + m1^2\,p1x + m1m2\,p1y + m1m3\,p1z) = 0$$

[Equation 142]

$$\frac{\delta d^2}{\delta t2} = 2t2(1-m2^2) - 2m1m2t1 - 2m2m3t3 + 2(-p1y^2 + m1m2\,p1x + m2^2\,p1y + m2m3\,p1z) = 0$$

[Equation 143]

$$\frac{\delta d^2}{\delta t3} = 2t3(1-m3^2) - 2m1m3t1 - 2m2m3t2 + 2(-p1z^2 + m1m3\,p1x + m2m3\,p1y + m3^2\,p1z) = 0$$

[Equation 144]

Rewriting Equations 142 to 144

$$2(-p1x^2 + m1^2\,p1x + m1m2\,p1y + m1m3\,p1z) = 2t1(1-m1^2) - 2m1m2t2 - 2m1m3t3$$

[Equation 145]

$$2(-p1y^2 + m1m2\,p1x + m2^2\,p1y + m2m3\,p1z) = 2t2(1-m2^2) - 2m1m2t1 - 2m2m3t3$$

[Equation 146]

$$2(-p1z^2 + m1m3\,p1x + m2m3\,p1y + m3^2\,p1z) = 2t3(1-m3^2) - 2m1m3t1 - 2m2m3t2$$

[Equation 147]

I matrix form:

$$\begin{bmatrix} 2(-p1x^2 + m1^2\,p1x + m1m2\,p1y + m1m3\,p1z) \\ 2(-p1y^2 + m1m2\,p1x + m2^2\,p1y + m2m3\,p1z) \\ 2(-p1z^2 + m1m3\,p1x + m2m3\,p1y + m3^2\,p1z) \end{bmatrix} = \begin{bmatrix} 2(1-m1^2) & -2m1m2 & -2m1m3 \\ -2m1m2 & 2(1-m2^2) & -2m2m3 \\ -2m1m3 & -2m2m3 & 2(1-m3^2) \end{bmatrix} \begin{bmatrix} t1 \\ t2 \\ t3 \end{bmatrix}$$

[Equation 148]

It can be noticed that Equation 148 has the following form:

$$\mathbf{K=MT}$$

[Equation 149]

Now, if it is wanted to find the point **T** which its distance to N lines is minimal, the equation 149 is found for every line and a total matrix system is found as follows:

$$\mathbf{K_T = K_1 + K_2 + \ldots + K_N}$$

[Equation 150]

$$\mathbf{M_T = M_1 + M_2 + \ldots + M_N}$$

[Equation 151]

$$\mathbf{K_T = M_T T}$$

[Equation 152]