THE EVALUATION AND IMPROVEMENT OF 3D SHAPE ACQUISITION TECHNIQUES

BY

HECTOR SANCHEZ SANCHEZ

A thesis submitted to the Faculty of Graduate Studies
In Partial Fulfilment of the Requirements for the Degree of

MASTER OF SCIENCE

Department of Mechanical and Manufacturing Engineering
University of Manitoba
Winnipeg, Manitoba

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

The Evaluation and Improvement of 3D Shape Acquisition Techniques

BY

Hector Sanchez Sanchez

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

Of

MASTER OF SCIENCE

Acknowledgments

Firstly, I want to thank Dr. Q. Peng, my advisor, for all the help he gave me along the time I was under his guidance. Surely without his knowledge, ideas and patience, this work cannot be finished.

Secondly, I want to thank Dr. D. Strong for his advice and commentaries related to this work. His idea of projecting a mesh and the use of a camera were great help for accomplishing this effort.

I also want to thank all my friends in the Research and Virtual Manufacturing Centre at the University of Manitoba. For all the advices, ideas, support and friendly relationship, thanks to Chulho Chung, Chunseng Yu, Luan Tao, Heewan Lee, Muhammad Ahsan. Special thanks to Wensong Hu, who helped me when I was jammed in the research. Thanks to Cid Flores to stay with me when I most needed. Her support, consideration and love make me feel great all the time I am with her.

Who deserve the greatest acknowledgment are my family. Without them, I would never have been able to even study a B.Sc. in the first place. My mother, father and brother, through their love, support and encouragement, helped me see that I had talent to achieve all what I proposed to do. Thank you very much.

Abstract

Three-dimensional (3D) shape acquisition has many important applications in different fields. Engineering, medicine, architecture, and entertainment are some examples where it is used.

This research is focused on the evaluation and development of a feasible and economic technique for 3D data acquirement. Nowadays different methods are used for 3D shape acquisition. These methods can be divided into the contact and non-contact approaches. The most popular contact devices are the joined arms and the coordinate measure machines (CMMs). The most common non-contact products are the laser scanners. These methods are high accurate and easy to use, but they are expensive.

Other methods used for 3D acquisition involves projected light into the object's surface and digital cameras for capturing the image. These methods are characterized by the low cost since only a projected light and a digital camera are needed.

Three objectives are proposed for this work. The first one is the evaluation of existing 3D acquisition methods. The second one is the implementation of a new approach to obtain confident 3D data at a low cost. And the last one deals with the comparison between the different approaches. To achieve the goals two methods are proposed based on image processing techniques. The methods are the shape from shading method (SFS) and the structured light approach.

An improved SFS approach is implemented based on one of the studied methods. A new structured light method is developed. In this method a grid with separation between lines of 6 pixels and 1 pixel thickness of line is projected to the object. A digital camera that captures the distortion of the grid takes the image. 3D data is then collected from the distortion of the grid. The 3D data is then recovered based on the distortion of the grid on the object.

Table of Contents

List of figures

List of Tables

Chapter 1. Introduction

Three-dimensional (3D) shape recovery is a technique used to obtain 3D data of physical objects. The applications of this technique are related to not only engineering processes such as reverse engineering, design, manufacturing and virtual environment creation, but also architecture, entertainment and medicine. The use of 3D digital images is becoming essential and necessary in those areas. Different products that can measure 3D shape of objects are now available in the market with different characteristics and prices. These products are either hard to operate or expensive to use. Therefore it is necessary to develop a new technique for the faster, easier and cheaper digitalization of 3D objects for industrial applications.

1.1 Research Background

The idea of the research came from a local foot clinic in Winnipeg. The clinic requires an automatic system for 3D foot shape acquisition. This research is intended to find a cost-effective solution to substitute the plaster molds currently used to obtain the foot shape for the clinic.

The requirements are analyzed based on application environments of the project. First of all, the proposed method has to be affordable by the clinic, be easy operation and quick data capturing process. Consequently, the research focuses on the evaluation and development of a feasible and economic technique for 3D data acquirement.

In order to have a good overview of the existing techniques, different 3D recovery approaches are evaluated. An existing method is improved and a new method is developed to improve existing techniques.

1.2 Objectives

The main objective of this research is to search for a cost-effective solution for 3D shape acquisition method. Although there are different approaches available to acquire confident 3D data, the cost of using them is high and the use of these methods is time-consuming. The examples include 3D shape measuring methods using Coordinate measuring machines and 3D lasers canners. In order to develop a new approach to simplify and economize existing 3D acquisitions techniques, following objectives are proposed for this research:

- Evaluate existing 3D acquisition methods.
- Implement a new approach to obtain confident 3D data at low cost.
- Compare the method developed with existing approaches.

1.3 Proposed methods

Two methods were proposed to acquire confident 3D data. One is the improvement of an existing method. Modification in image processing was applied to obtain better results. Another is a new method, which combines the structured light and stereo techniques. This method reconstructs 3D shape using only a camera and a projector. For the existing stereo approach, minimum two cameras are required for the reconstruction process. In the proposed method, a projector is used to substitute one camera. Even though the proposed

method is similar to the structured light approach, difference is that a pattern is projected on the object. The projected pattern is a grid, not a strip. The pattern is fixed; there are no lines or strips projected onto the object in sequence in a time period. Other difference is number of images needed. In the proposed method, only one picture is required to reconstruct the object.

In the operation to capture a 3D object, a camera takes a picture of the object on which a grid is projected; the deformed pattern on the image gives the clue to reconstruct 3D points of the object. The related coordinates of the grid are pre-defined, the location of each point to recover is known. The height is found with triangulation. With this method it is intended to get a good 3D approximation of the object under study.

1.4 The structure of the thesis

The thesis is organized as follows: A description of different 3D shape recovery methods is presented in Chapter 2. The comparison of these systems is presented. The practicability of each method is also discussed, which forms the basis of the thesis work. Two of the methods are selected for further work based on the objectives and restrictions established for the research. These methods are the shape from shading (SFS) approach and the structured light method.

Chapter 3 presents an overview of SFS methods. These methods are analyzed and tested. Finally an improved technique based on one of the chosen algorithms is presented. A comparison with the other methods is performed.

The structured light approach is reviewed in Chapter 4. Different research that has worked in this area is presented. The purpose of this chapter is to review existing research in structured light approach and propose a feasible method. At the end of the Chapter, a new method is proposed using structured light and triangulation.

Chapter 5 introduces the development of a new method. The components of the system are presented. The calibration procedure is solved. The completed 3D reconstruction system is discussed. All the algorithms related to the proposed method are described. These include the corner detection, interpolation, relation between coordinates of the equipment used, triangulation, and mesh generation. The algorithms are programmed using Matlab, and developed codes are presented in the Appendix section.

The results obtained by the proposed method are presented in Chapter 6. A comparison with a laser scanner is performed and discussed.

Finally, the conclusions and further work of the research are presented in Chapter 7.

Chapter 2. Review of 3D acquisition Techniques

3D shape recovery techniques are used to obtain 3D data of a physical object for a specific application. According to methods of data acquisition, these techniques can be classified in two groups: contact and non-contact techniques. The principal difference between them is the way of the shape acquisition. The related-techniques are shown in Fig. 2.1.

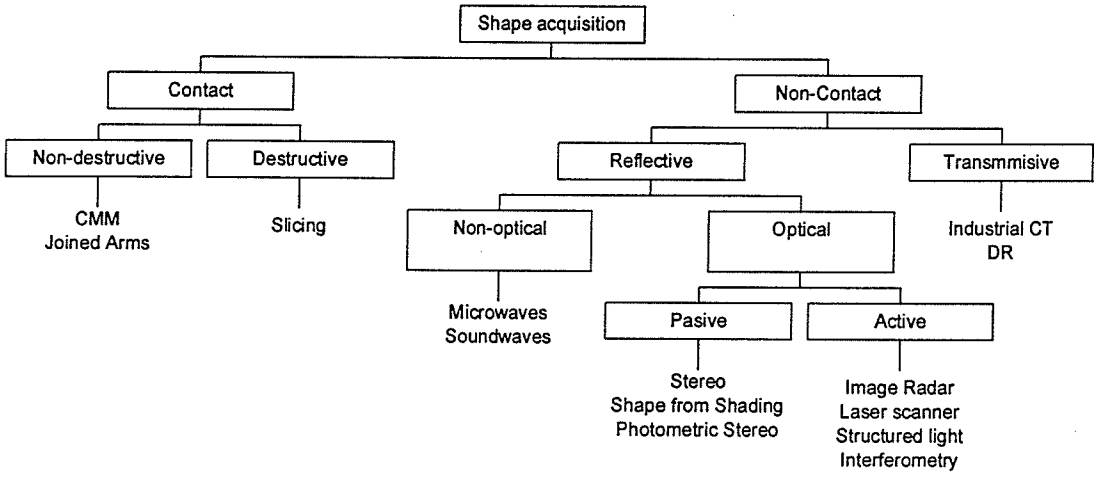


Fig. 2.1 3D recovery classification

In contact techniques a physical mechanical component follows the contour of the part of interest. Two different ways of data acquisition are used: destructive and non-destructive techniques.

Destructive devices are those that destroy the part during the process. The system uses a precise fly cutter to accurately machine away ultra-thin layers of the part. The location of

each layer is computed by software that automatically files each point of data. Because the devices cut the part into pieces, the part becomes useless for further applications.

Non-destructive methods leave the part intact so it could be used again. In this method a mechanical arm is used for data acquisition. It follows the contour of the part sending the detected signal to a computer that registers the coordinates. A coordinate measuring machine (CMM) is an example of this method.

The measuring device does not touch the part of interest using non-contact techniques. Usually a source of energy such as laser, light, microwave or sound, is used to create a reflective effect that is captured by a camera or other data acquisition systems. These methods are divided in two areas: tramsmisive and reflective. In tramsmisive techniques light passes through the object. Computer topography (CT) and digital radiography (DR) are examples of this technique. In reflective techniques the energy applied is reflected from the object so that a device is able to capture the signal and process it. Depending on the energy applied, it can be sub-classified into optical and non-optical techniques. In the non-optical method a frequency energy source such as microwaves or sound waves are projected to the object. This method is based on the Doppler effect considering the energy that is reflected from the surface.

The optical method uses lights as energy. Depending on the light applied, it is classified into passive and active technique. In the passive technique no external energy is imposed onto the object of interest. The ambient light is enough to get the shape of the object. One

of the principal problems that these techniques face is at the moment of data reconstruction. The process of matching the same point at different captures is complex; the difficulty includes repetitive patterns, occlusion and bland regions. The advantages are related to redundancy of images, full modeling of all parameters and the low cost. The disadvantages are the high level of computation required for the surface reconstruction; the setting up for each required image; the lack of techniques for fast and dynamic measurement and low accuracy. Some examples of the passive technique are stereo image acquisition, shape from shading (SFS), photometric stereo (PS), and photogrammetry, where commercial digital cameras, charge-coupled device (CCD) cameras or video cameras are used for the data acquisition.

Active techniques are based on an emitter that projects some sort of structured and controlled illumination to the object of interest. In this approach light, such as point or stripe patterns are projected onto the surface. A sensor, typically a CCD camera, acquires images from the distortion of patterns reflected by the object surface. Depth information is reconstructed by triangulation. Laser scanners are the most common and popular examples. Some of the advantages of this technique are related to the accuracy obtained and the easy implementation they can provide. The principal disadvantage is related to the noise that the ambient light can produce and the high cost. [1-3]

In this chapter a review of the most representative 3D acquisition techniques are presented. The methods studied are the laser scanner, CMM, SFS, structured light, and stereo. The characteristics of each technique and their working processes are discussed.

After analyzing the advantages and disadvantages of each of the presented techniques, a new approach is proposed.

2.1 Laser Scanners

Laser scanning is an active stereoscopic technique where the distance of the object is computed using a directional laser light source and a camera. The working principle is as follows. A laser beam is deflected from a mirror onto the desired object. Light is reflected and captured by a CCD camera located at a known distance from the laser. Data is then estimated by triangulation. The camera records the deformation at each point based on the reflected light, and creates a digitized image of the object known as a cloud of data. This process is applied more than once to cover the complete surface and to fill the holes that the initial scan leave. The separate data files from each scanning are combined to produce a complete integrated image of the scanned object. This method is the most common technique for digitalization because it is fast, robust and easy to use compared to other methods. The amount of data acquired makes the system highly accurate. The cost of this device is high.

2.2 Coordinate Measure Machines (CMMs)

CMMs are high precision measurement equipment that integrates mechanical, optical, numerical control and computer technologies. Liu et al explained the principle of operation of a CMM known as parallel-link CMM [4]. The CMM consists of a structure of three perpendicular joints with servomotors, struts, a probe and a platform where the object is placed. When the process is started, the servomotors, located in the three joints,

control the struts to expand and contract along their own axis direction. Variation in the length of the struts makes the platform move in a 3D space. The probe that has a ball in the end does the measurement of the object. Data acquisition is performed in two perpendicular directions registering $n \times m$ coordinates. Where n and m are the number of measuring points. The distance between two measurement points may be constant for planar surfaces or adapted to a variable curve surface. The CMMs require calibration. The ISO10360-2 standard is used to acceptance of re-verification of the machine [5].

Brenner et al discussed CMM limitations including the high cost and low measurement speed, corresponding to a long validation time [6]. CMMs are used as the central activity in quality control and reverse engineering. Quality control is focused on tolerance verification, which is based on tolerance analysis and verification algorithms. In reverse engineering CMMs are able to reconstruct the shape of an object by meaning of CAD software. CMMs provide an accurate and flexible method for data acquisition. Its advantages are diminished by the slow rate of data captured that is related to the speed that the probe is moved between measurements.

2.3 Shape from Shading (SFS)

SFS is the technique used in computer vision to derive a 3D scene description from one or more 2D images. The first reference to SFS is dated back to Horn in 1970 who formulated the reconstruction problem [7]. He established that given an intensity image of a continuous surface, for which the reflectance and illumination are known, the surface can be recovered. Therefore, SFS deals with the recovery of shape form a gradual

variation of shading in the image given the light source and the surface reflectance information. This method is cheap and easy to implement. No expensive lights are required; a simple flashlight can be used for the reconstruction process. A camera and a source of light are required for the reconstruction. The setup of the system is hard since the light has to be located in a way that illuminates the object. Surface reconstruction depends on the way that the light projected onto the object. The surface reflectance produces noise in the reconstructed image. The accuracy obtained by this method is low.

2.4 Structured Light

Structured light (SL) is a technique based on the projection of light onto the object of interest. Different authors [8-12] used this technique widely for various applications in robotics and computer vision. Characteristics of the projected light, such as colour, dimension and location, have to be known in order to solve the correspondence problem. When the camera captures the stripe, its characteristics allow obtaining a unique match. More than one image of the scene is required to reconstruct the entire scene. This is required since several strips are projected in different times with different intensities and characteristics. The most common example is the sequence of binary stripe patters projected by a projector and recorded with a CCD camera. The patterns captured in the image as deformed stripes are detected with a suitable image algorithm. Thus, the sequences of patterns are converted into sequences of bit planes. In other words, the location of a specific object point is encoded in a sequence of pixel values according to the bit plane. Given the position of the illuminant projector and the camera, the range data of the object can be computed by trigonometry. This method is cheap and easy to

implement. The characteristics of the projected pattern solve the correspondence problem, making the method suitable for high volume reconstruction. A dark room is required to reduce the noise produced by light reflection of the object's surface. Since different patterns are projected one after another, the method is slow and the algorithm is complex.

2.5 Stereo

This method is based on the way of the human visual system to perceive the depth of a scene. It is known as stereovision. Stereovision refers to the process in which a scene is watched by the right and left eye. For example, for two image frames, a projection carries a typical point of the scene onto two different image locations. The difference is called the disparity. From this disparity and from the knowledge of some basic geometric relations between the two imaging systems, the depth of the point can be calculated.

A particular stereo reconstruction process consists of three stages. The first is known as feature detection that locates special feature points in a scene. In the second step, the feature points have to be matched between images, allowing the calculation of a disparity map of the scene. Finally, in the third step, the depth values of these points are calculated from the disparity map. The stereo process is very similar to photogrammetry, except for the feature-matching step, which is usually much simpler in photogrammetry as the features are marked points fixed on the object surface. The feature detection can rely on results of the vision research field, and the depth calculation is based on geometry, the matching of feature is commonly believed to be the most difficult and time-consuming

step of stereovisions. Thus, most of the research in stereovision is in the field of feature detection and matching.

2.6 Analysis of the techniques

The analysis is to evaluate the different characteristics of the presented techniques. The goal is to define the method to follow or establish a new idea to develop. Table 2.1 presents the comparison between systems and their advantages, disadvantages and cost.

Since one of the restrictions established for this research is to develop a fast and non-expensive method, the CMM and the laser scanner are discarded. Because another restriction is to use only one digital picture to make the system cheaper, the stereo method is not considered. Therefore the research is oriented to those techniques where only one picture and an external light source are required. For that reason, the SFS technique and the SL approach are considered.

Table 2.1 Comparison between 3D acquisition systems

Type	Technique	Advantages	Disadvantages	Cost
CONTACT	Slicing	-The volume is known. -High accurate	- Part is useless -Takes long time -No colour information	High
	Mechanical Probe	-Specific attention on important points -More measurements could be done -High accurate	-Takes long time -Person has to move it -Probability of mistake -No colour information	High
NON CONTACT	Stereo and Photogram.	-Infrastructure is cheap -Able to sense the complete object -Colour information	-Slow -Advance error detection algorithm -Camera Workspace	Low
	Shape from Shading	-Infrastructure is cheap -No expensive external light	-Slow -Surface reconstruction is hard -Regular accuracy -No colour information -Noise from the light condition	Low
	Structured Light	-High quality precision method -High Accuracy -Easy to use -High working volume	-Surface Reflectance -Noisy fringes -Requires image enhancement -Slow data acquisition -No colour information -Requires specific light conditions	Medium
	Laser Bean	-Short scanning time -Not requires specific light conditions -Good resolution -Low measurement noise -High accuracy -No distortion -Easy to use -High working volume	-Considerable post processing data	High

Chapter 3. Shape From Shading (SFS)

SFS is a technique used in computer vision to derive a 3D scene description from one or more 2D images. SFS deals with the recovery of shape form a gradual variation of shading in the image given the light source and the surface reflectance information.

In this section a description of the basic theory for SFS is covered, followed by a comparison of different SFS techniques and finally an improved method is proposed and tested.

3.1 SFS Basic Theory

An image is defined as a 2D pattern of brightness produced in an optical forming system [13]. Brightness is referred to two different concepts: image brightness and the scene brightness. Image brightness, known as irradiance (E), is related to the energy flux that reaches the image plane as shown in Fig. 3.1. Scene brightness, known as radiance (L), is the reflected energy by the surface as shown in Fig. 3.2

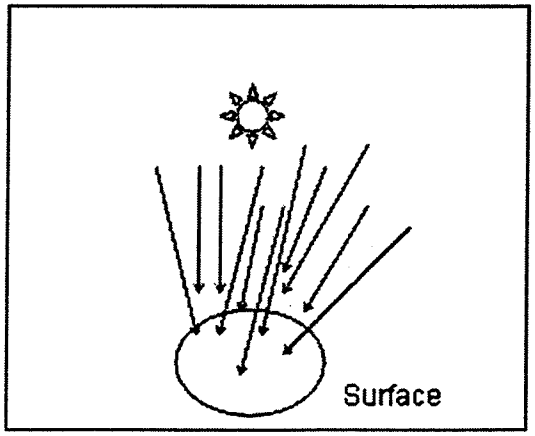


Fig. 3.1 Irradiance

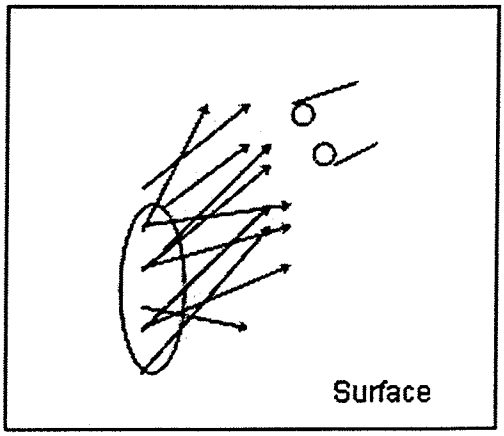


Fig. 3.2 Radiance

The tilt (τ) and the slant *(σ)* angles define the direction of the light reaching the object. Tilt of the illuminant is the angle between the projection of the surface normal on the image plane and the horizontal axis. Slant is defined as the angle between the line of sight and the surface normal [14,15]. The representation of these angles is shown in Fig. 3.3

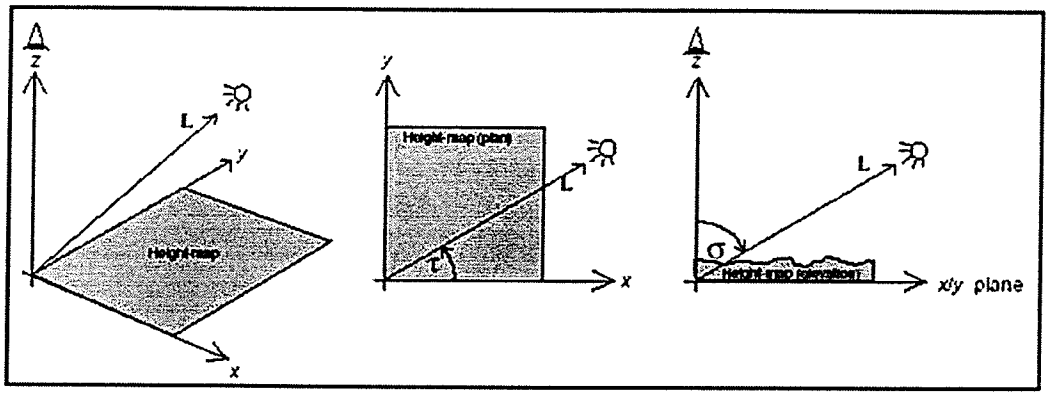


Fig. 3.3 Representation of Tilt and Slant angles

The bi-directional reflectance distribution function (BRDF) tells how bright a surface appears when it is viewed from one direction while light falls on it from another. Once the BRDF is known, it is possible to calculate the reflected radiance in terms of the source radiance.

Based on the physical properties different surfaces can be classified from the light reflection. Surfaces can be categorized as pure Lambertian, pure specular, and the combination of both known as hybrid surfaces. An ideal Lambertian surface is the one that appears equally bright from all directions and reflects all incident light, absorbing none [16]. The Lambertian model is the simplest model of image formation.

A specular surface is presented when the incident angle of the light source is equal to the reflected angle. Finally, hybrid surfaces are combination of Lambertian and specular surfaces. Lambertian surface is considered in this research.

Surface orientation is another important parameter to consider. Knowing the position of a surface, it is easy to determine other constants such as the reflectance map. To explain this concept a smooth surface is considered since it has a tangent plane at every point. This tangent represents the orientation of the surface at that point. The normal of the surface is known and it is used to specify the orientation of the plane. The coordinate system is fixed to an axis aligned with the optical axis of the imaging system. The surface normal is aligned in terms of z and the partial derivatives of z with respect to x and y axis. If a small step δx is considered to start from a given point (x, y), the change in z can be represented using Taylor series expansion as

$$\delta z = \frac{\partial z}{\partial x}\delta x + e \tag{3.1}$$

where e contains higher order terms. p and q are used to represent the first partial derivative of z with respect to x and y, respectively. Therefore, p is the slope of the surface measured in the x-direction, while q is the slope in the y-direction. If a small step of length δx in x direction is taken, the height changes by $p\delta x$. Similarly for δy in y direction, the height will change by $q\delta y$. The relation between p and q is shown in Fig. 3.4.

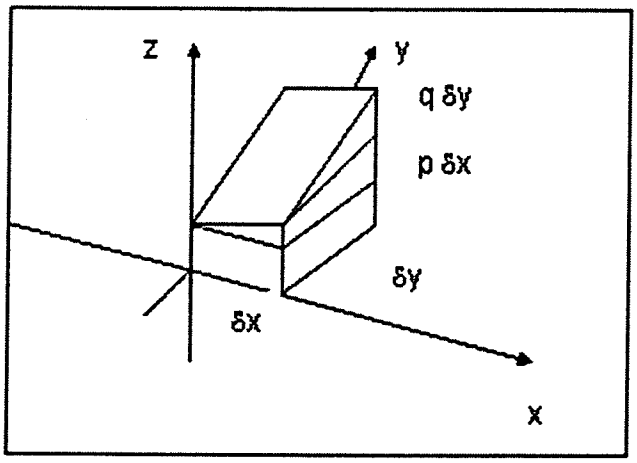


Fig. 3.4 Surface orientation is parameterized by the first partial derivatives p and q of the surface height z

The reflectance map makes explicit the relationship between surface orientation and brightness. It contains information about surface reflectance properties and light source distributions, giving scene radiance on surface direction. When a source with radiance E illuminates a Lambertian surface, the scene radiance is described by

$$L = \frac{1}{\pi} E \cos\theta_i \tag{3.2}$$

for $\theta_i \geq 0$ where θi is the angle formed between the surface normal and the direction toward the source. That equation can be represented in unit vectors as follows:

$$\cos\theta_i = \frac{1 + p_s p + q_s q}{\sqrt{1 + p^2 + q^2}\sqrt{1 + p_s^2 + q_s^2}}. \tag{3.3}$$

The last equation shows how bright an image is depending on the surface orientation. The result of the equation is called the reflectance map and it is denoted by $R(p,q)$.
Usually the reflectance map is normalized to set its maximum to one since the image irradiance is proportional to the fixed brightness of the source. A Lambertian surface

illuminated by a single distance point source is related to:

$$R(p,q) = \frac{1 + p_s p + q_s q}{\sqrt{1 + p^2 + q^2}\sqrt{1 + p_s^2 + q_s^2}}. \tag{3.3}$$

The lighting conditions are traditionally simplified assuming an infinite point light source. This is equivalent to say that the light rays are parallel and reach each point of the surface with the same angle.

3.2 SFS Techniques

According to Tsai et al [17] and Kimmel et al [18], recovering shape can be expressed in four different ways: depth, surface normal, surface gradient and surface slant and tilt.
In depth the z plane is supposed to be recovered considering the relative surface height about the $x-y$ plane. In the surface normal the orientation of a perpendicular vector to the tangent plane on an object surface is recovered. In the surface gradient the depth rate of change is considered in x and y coordinates. The surface tilt and slant are related to the surface by a magnitude of the surface normal.

Different authors divided SFS into four groups according to the algorithm used: minimization, propagation, local and linear approaches [19-21]. Minimization and propagation approaches are generalized as global approaches. Global approaches propagate information across a shaded surface starting from points with known surface orientation. Global algorithms assumed that the surface is smooth in some sense in order to extract surface orientation. Local and linear algorithms attempt to estimate shape from local variations in image intensity as was established by Pentland [15]. Local algorithms

suppose surface shape assumptions in order to extract surface orientation from the shading information within a small image neighbourhood. In the following sections the analysis of these techniques are discussed.

3.2.1 Minimization approaches

Minimization approaches recover shape by minimizing an energy function. The function involves one of the following constrains: a) Brightness constraint, which is derived directly from the image irradiance. b) Smoothness constraint that ensures that the surface is smooth in order to stabilize the convergence to a unique solution. c) Intensity gradient constraint that requires that the intensity gradients of the input image and reconstructed image are in the same x, y directions, and d) unit normal constraint that forces the recovered surface normal to be a unit vector [19, 20]

3.2.2 Propagation approaches

In propagation approaches the shape information is propagated from a set of surface points to the entire image. Starting points are usually singular points where the intensity is either maximum or minimum. At singular points the shape of the surface is either known or can be determined.

3.2.3 Local approaches

The basic assumption in these approaches is that the surface is locally spherical at each pixel point. Pentland presented a solution in 1984 [22], and then an improvement of it in 1988 [15]. He assumed that surface points are umbilical points. The idea was to construct

a linear approximation to the true reflectance function, allowing efficient close form solution for the surface shape. In order to obtain a convenient and efficient solution, the intensity equation was transformed into the Fourier domain.

3.2.4 Linear approaches

Pentland used the linear approximation of the reflectance function in terms of the surface gradient (p,q) [23]. Fourier transform to the linear function was applied to get a closed form solution for the depth at each point. The depth map can be computed rearranging the terms in the equation and taking the inverse Fourier transform. Tsai and Shah applied a discrete approximation of the gradient (p,q) using finite differences first, and then employed the linear estimate of the reflectance function in terms of the depth directly [17]. This approach makes the algorithm faster than other algorithms because each operation was purely local and it could be applied to any reflectance function.

3.3 Selected Algorithms

Due to the good results reported by the authors and the easy implementation of the algorithms, Pentland's [23] and Tsai and Shah's approaches [17] were selected and programmed in Matlab. Pentland's method was chosen because the algorithm produces good results on most surfaces that change linearly. It was also found that this approach produces the best result for real images. Tsai and Shah's approach was selected due to the simplicity of the algorithm and the good results it generated in a short time.

These two algorithms were further tested with synthetic and real images. Different synthetic images of a ball were created in "Autodesk VIZ 4" with different illumination conditions as shown in Fig. 3.5. Fig. 3.6 shows the synthetic image that generates the best result when the 3D surface was reconstructed. This image is considered as the ideal illumination model due to the results. The image has a bright illumination in the centre that gradually becomes dark. This is the desired image for any SFS algorithm.

For the real images a camera and lights were required to get the pictures. The camera used is a "Logitec QuickCam" with max resolution of 640X480 pixels. The lights were outdoor bulbs arranged in such a way that the object gets the illumination in the required place. Several bulbs were distributed around the object to obtain a better illumination as shown in Fig. 3.8. The distribution of the lights was performed to get a better illumination model. Even though in theory the lights are punctual, at the moment of illuminating only with one lamp, the object was not suitable for reconstruction. Therefore, several lamps were located along the object to obtain a better distribution of light in the entire object. A potentiometer was also used to vary the intensity of the light. The foot image is shown in Fig. 3.9.

The real images used in this section were a ping-pong ball, and a foot. The ping-pong ball was subject to different light conditions as seen in Fig. 3.7. A rubber cover was added to the ball to make it a lambertian surface. Without the cover, the image behaves like a specular surface, and the light was reflected producing noise in the reconstruction. For this model several different illumination conditions were applied to it, only the best configuration is shown.

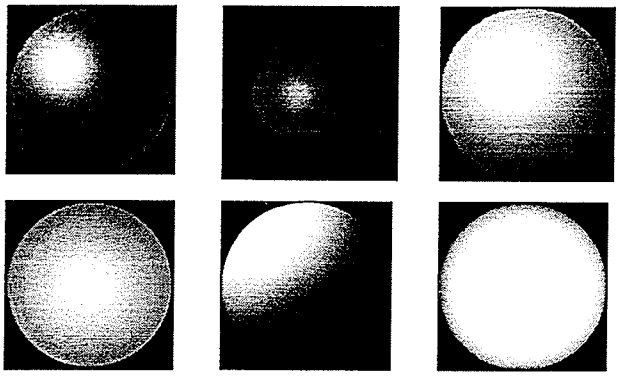


Fig. 3.5 Synthetic sphere images with different light conditions created in Autodesk VIZ 4

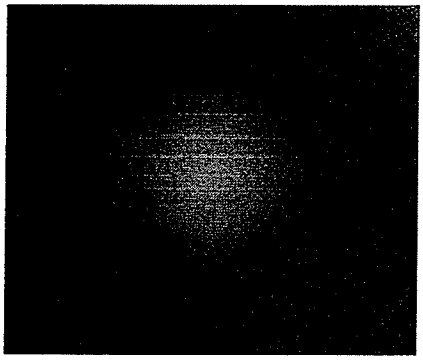


Fig. 3.6 Ideal sphere illumination

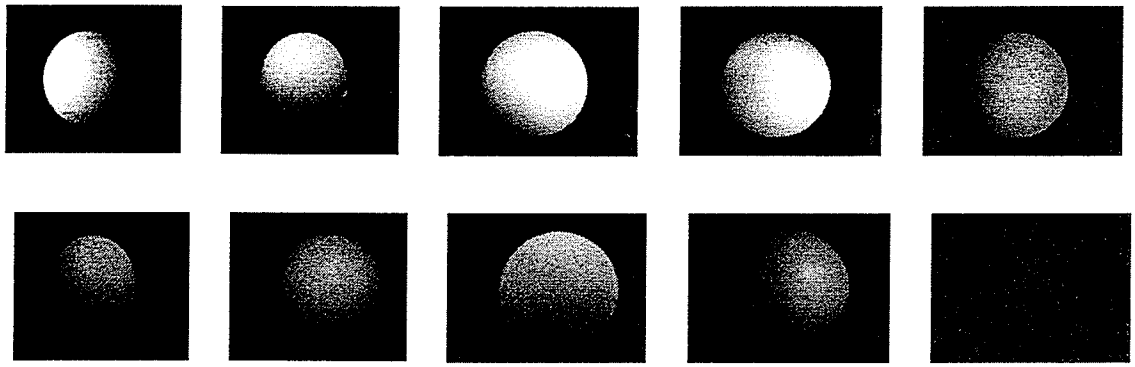


Fig. 3.7 Real sphere image with different illumination

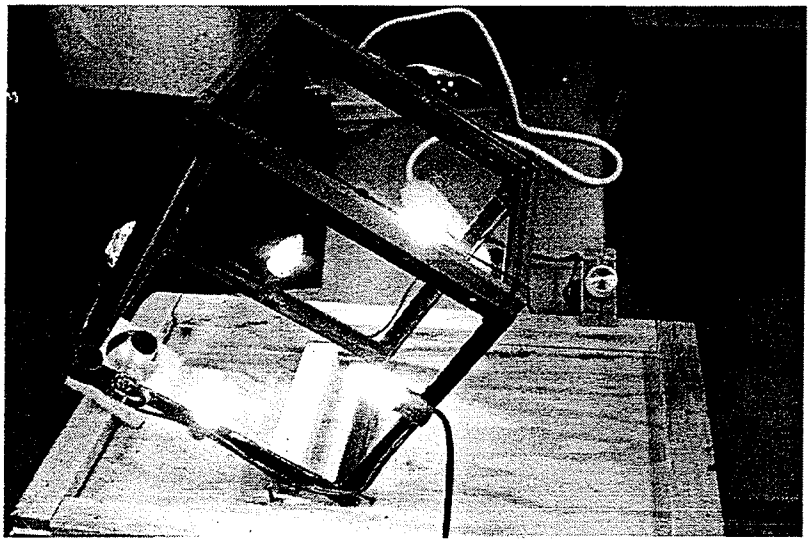


Fig. 3.8 Lights arrangement in the SFS implemented method

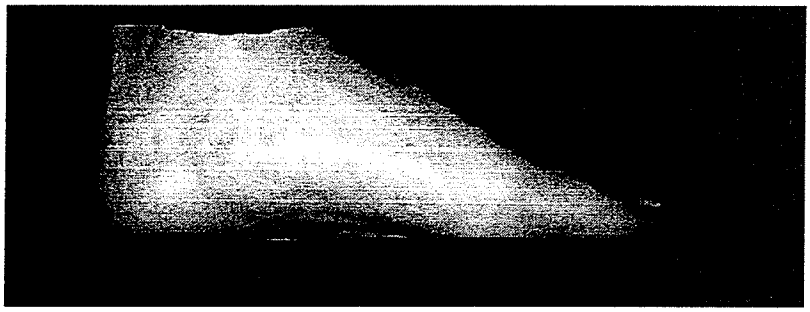


Fig. 3.9 Real foot image

3.3.1 Pentland Approach

Pentland's algorithm uses a linear approximation of the reflectance map. The assumptions for this method are related to the image properties [23]. The surface is assumed to be Lambertian, the surface is illuminated by distant point sources and the surface does not produce shadows. Under orthography projection the normalized image intensity is: $I(x,y) = \frac{p\cos\tau\sin\sigma + q\sin\tau\cos\sigma + \cos\sigma}{\sqrt{p^2+q^2+1}}$. (3.4)

A Taylor series expansion is applied to convert the equation to a form that relates the image and 3D surface in terms of their Fourier transforms as:

$$I(x,y) = \cos\sigma + p\cos\tau\sin\sigma + q\sin\tau\sin\sigma - \frac{\cos\sigma}{2}(p^2+q^2). \quad (3.5)$$

This is a good approach if p and q are small and the illuminant direction is oblique. If the illumination is vertical, the quadratic terms in the last equation dominate and a frequency doubling effect occurs. Now the Fourier spectrum $F_z = (f,\theta)$ of $z(x,y)$ is:

$$F_z(f,\theta) = m_z(f,\theta)e^{i(\phi_z(f,\theta)} \quad (3.6)$$

where $m_z(f,\theta)$ is the magnitude at position (f,θ). Then the Fourier transform of the image is:

$$F_I(f,\theta) = 2\pi \sin\sigma f m_z(f,\theta) e^{i(\phi_z(f,\theta)+\pi/2)}\left[\cos\theta \sin\tau + \sin\theta \sin\tau\right]. \tag{3.7}$$

From this equation if the illuminant direction is given, the Fourier transform of the surface can be recovered directly as:

$$F_z(f,\theta) = \frac{m_I(f,\theta) e^{i(\phi_I(f,\theta)-\pi/2}}{2\pi \sin\sigma f[\cos\theta \cos\tau + \sin\theta \sin\tau]} \tag{3.8}$$

The Matlab code is presented in Appendix A. The illuminant conditions were assigned based on experimentation. It was seen that the better results in the scale were presented when: a) the values were close to one in the x direction, b) greater than one in the y direction, and c) z was set to one all the time. It was noticed that small values in the illuminant direction, generates the reconstructed surface to be bigger than the original object.

Fig. 3.10 shows the results of the synthetic ball images using Pentland's algorithm. It can be seen that the brightness part of the sphere in the picture has the maximum value in the reconstructed image. Unfortunately, the brightness section of an object sometimes cannot be reconstructed as the case of the first image in Fig. 3.10. The reconstructed object that more approximates to a 3D sphere is presented in Fig. 3.11. In this figure, light conditions applied to the object make the brightness part be in the center and gradually vanish along the object until black.

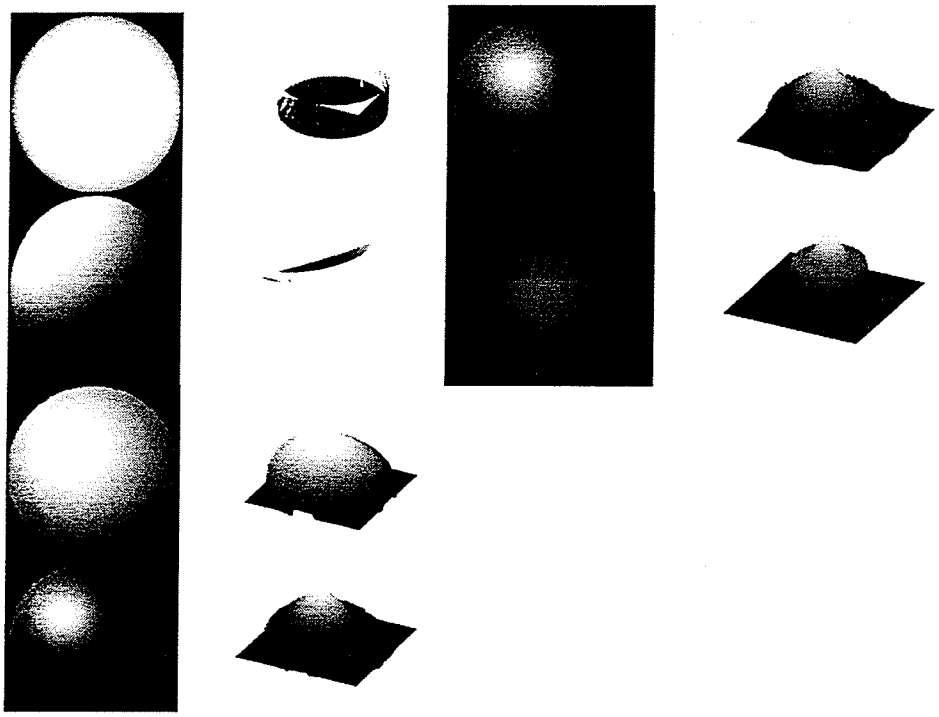


Fig. 3.10 Results of different synthetic ball images using Pentland approach

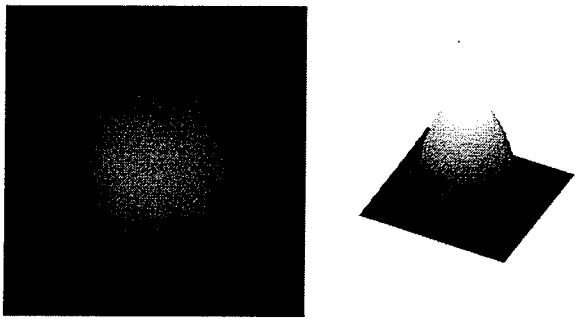


Fig. 3.11 Ideal illumination for SFS recovery in a sphere and the reconstructed shape using Pentland approach

Fig. 3.11 shows some of the results of the ping-pong ball under different light conditions, and Fig. 3.13 shows the reconstructed foot.

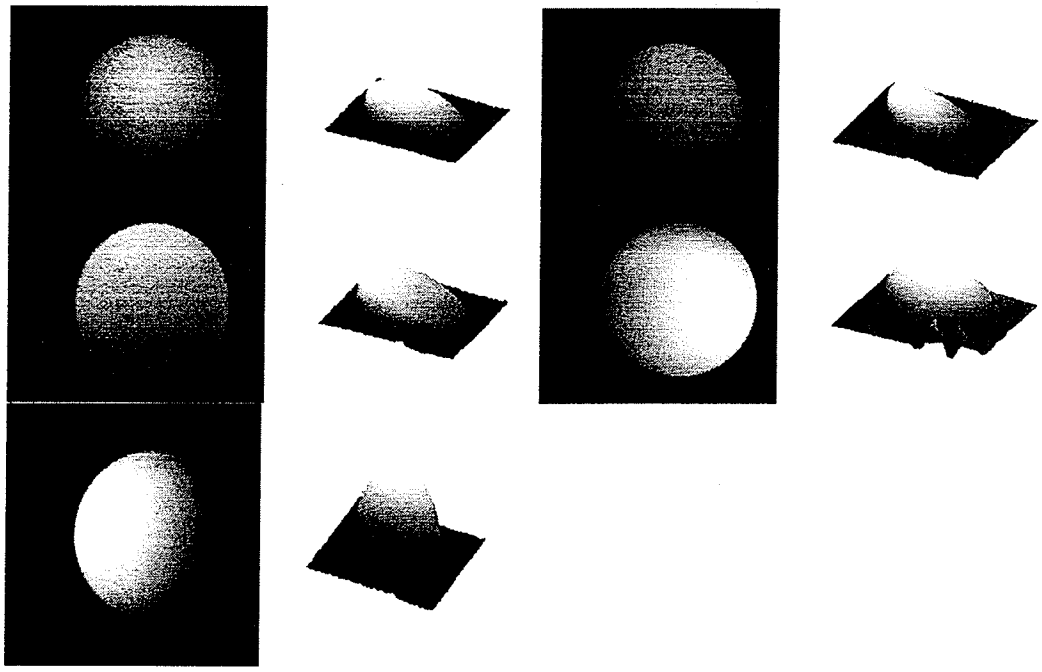


Fig. 3.12 Results of the ping-pong ball under different light conditions using Pentland approach

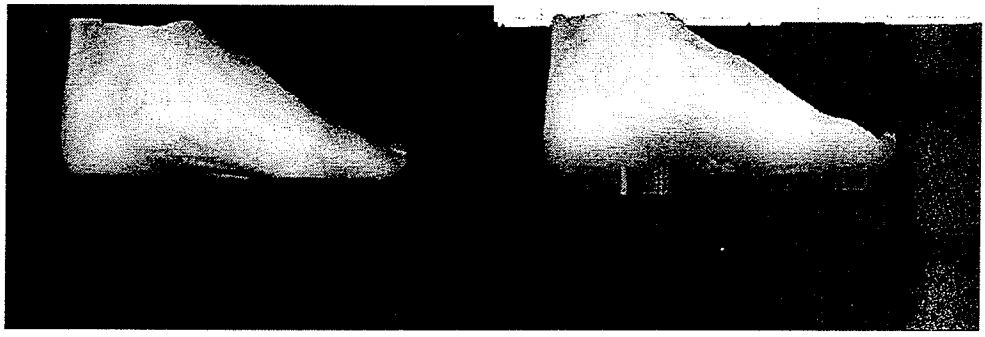


Fig. 3.13 Real Images reconstructed using Pentland approach

3.3.2 Tsai and Shah Approach

Tsai and Shah used the reflectance function for lambertian surfaces [17]:

$$E(x,y) = R(p,q) = \frac{1 + pp_s + qq_s}{\sqrt{1+p^2+q^2}\sqrt{1+p_s^2+q_s^2}} = \frac{\cos\sigma + p\cos\tau\sin\sigma + q\sin\tau\sin\sigma}{\sqrt{1+p^2+q^2}} \quad (3.9)$$

The above reflectance equation can be written as

$$\begin{aligned} 0 &= f(E(x,y), Z(x,y), Z(x-1,y), Z(x,y-1)) \\ &= E(x,y) - R(Z(x,y) - Z(x-1,y), Z(x,y) - Z(x,y-1)) \end{aligned} \quad (3.10)$$

For a fixed point (x,y) and a given image E, a linear approximation using Taylor series expansion up through the first order term of the function f about a given depth map Z^{n-1}. Followed by a Jacobian iterative method, the equation is reduced to:

$$0 = f(Z(x,y)) \approx f(Z^{n-1}(x,y)) + (Z(x,y) - Z^{n-1}(x,y))\frac{d}{dZ(x,y)} f(Z^{n-1}(x,y)). \quad (3.11)$$

Then for $Z(x,y) = Z^n(x,y)$, the depth map at the nth iteration can be solved as:

$$Z^n(x,y) = Z^{n-1}(x,y) + \frac{-f(Z^{n-1}(x,y)}{\frac{d}{dZ(x,y)} f(Z^{n-1}(x,y))} \quad (3.12)$$

where

$$\frac{df(Z^{n-1}(x,y))}{dZ(x,y)} = -1 * \left(\frac{(p_s + q_s}{\sqrt{p^2+q^2+1}\sqrt{p_s^2+q_s^2+1}} - \frac{(p+q)(pp_s+qq_s+1}{\sqrt{(p^2+q^2+1)^3}\sqrt{p_s^2+q_s^2+1}} \right) \quad (3.13)$$

This algorithm is easy to implement. It also produced good results. The illuminant light direction was tested with the values recommended by authors. The Matlab code is presented in Appendix B. Fig. 3.14 shows the results of the synthetic images. Fig. 3.15

shows the results of the ideal illumination. Fig. 3.16 shows the results of the ping-pong ball and Fig. 3.17 shows the result of the real picture of the foot.

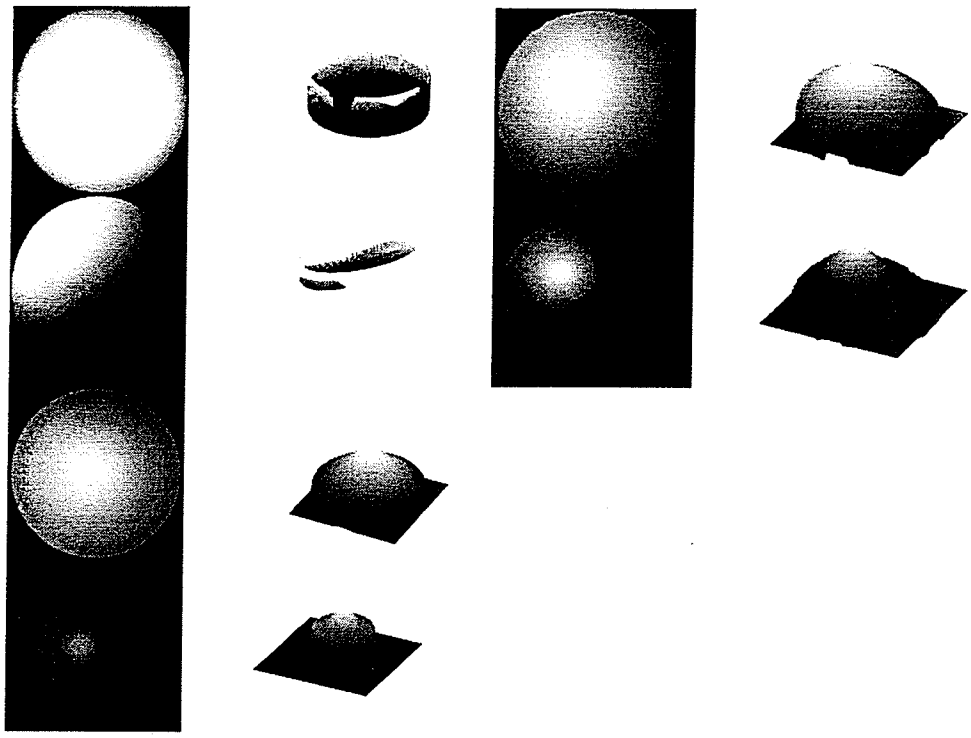


Fig. 3.14 Results of the synthetic images under different light conditions projection using Tsai and Shah's algorithm

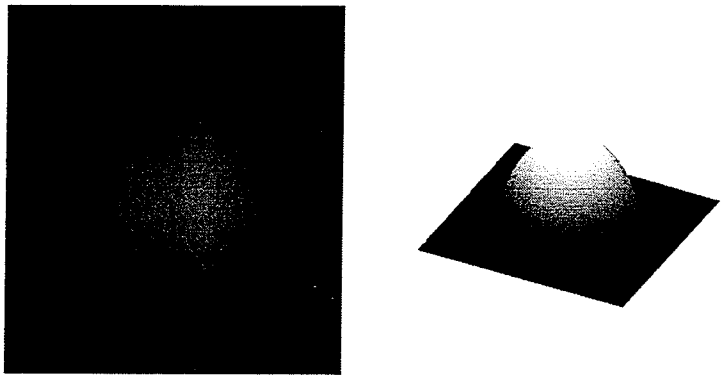


Fig. 3.15 Ideal illumination for SFS recovery in a sphere and the reconstructed shape using Tsai and Shah algorithm

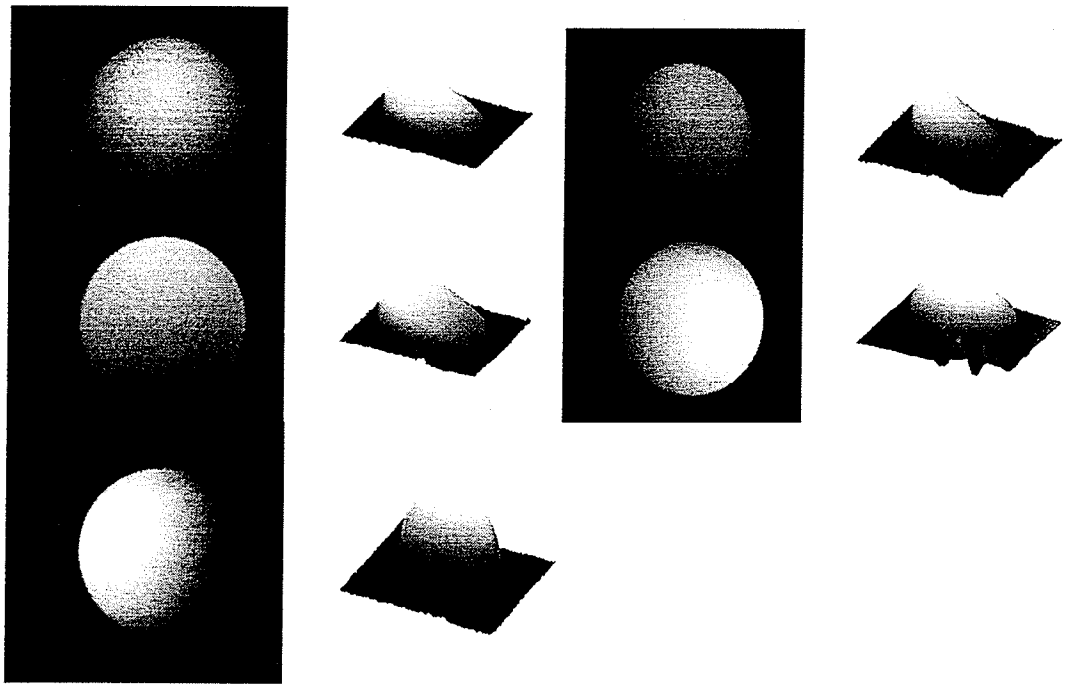


Fig. 3.16 Results of the ping-pong ball under different light projection using Tsai and Shah algorithm

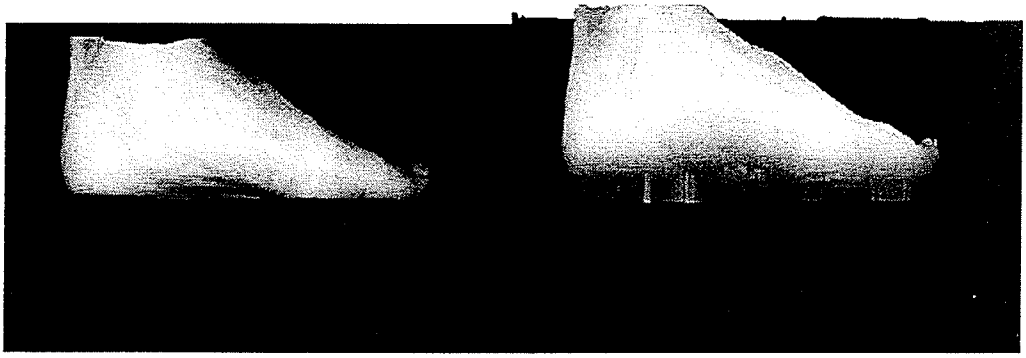


Fig. 3.17 Real Image reconstructed by Tsai and Shah's algorithm

The last set of pictures (3.10-3.17) show clearly that Tsai and Shah algorithm presents a better result than Pentland's method. The reconstructed images are smoother comparing the results in Fig. 3.11 and Fig. 3.15. Tsai algorithm presents a better approximation and the process is faster. The sphere looks compact and the surface looks better. In the ping-pong ball the results are not ideal due to the light conditions applied. The important

difference is shown in the foot image. It is clearly shown that Tsai and Shah's algorithm presents more detail than Pentland's algorithm. Therefore, this algorithm was chosen for further work.

3.4 Modified SFS algorithm

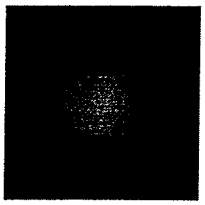
Tsai and Shah's algorithm was improved to obtain better results. The improvements were focused on the image processing. Before applying the algorithm an image threshold, image enhancement and image restoration were applied. A threshold was implemented to isolate the desired object by removing images that are not desired in the recovering process. A median filter was followed for smoothing the image by the suppression of noise, and preserving of edges. Finally, the image restoration was applied to remove or minimize some known degradations in an image such as noise infunded by the lights. Appendix C shows the modified algorithm. After applying these modifications, the quality of the resulting image increments and the reconstructed image is smoother. A comparison between the three algorithms is presented in Figs. 3.18-3.20. The illumination model was tested for each image. The bulbs located in different parts along the objects with different intensities were the only illumination present.

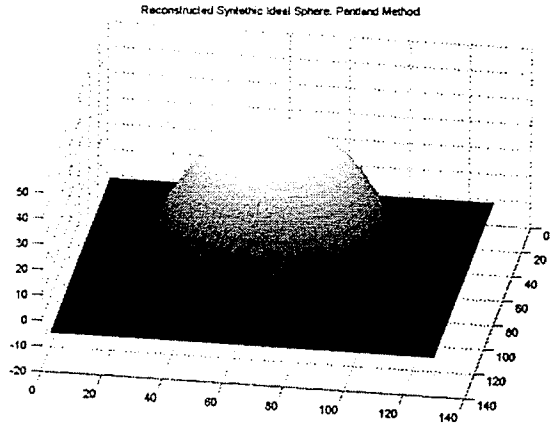
3.5 Discussion

Although the presented results show a 3D shape, the dimensions that the algorithm generates are not accurate enough as shown in Tables 3.1- 3.3. These tables compare the minimum value of the data acquired that is the lowest point of the height, and the maximum value registered that is the highest point obtained in the reconstruction process.

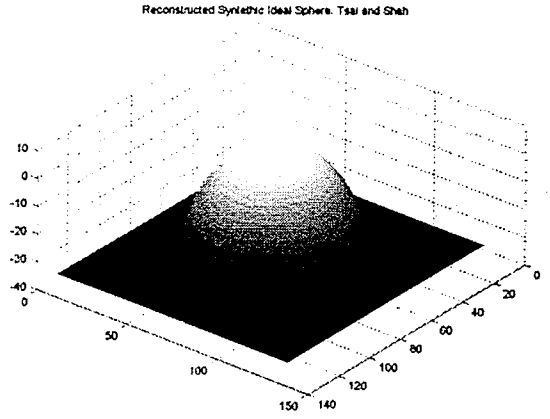
These tables also show the total difference between the values of the reconstructed image, the real difference between the measured image and the original image, the error between the true depth and the estimated one and finally the computer processing time. The computer used is a Pentium IV, with 1.7GHz CPU and 520 MB DDR memory.

As shown in the Tables 3.1-3.4, the values of the reconstructed images comparing with the original ones differ considerably. In the case of the foot recovery, the reconstructed high value is a good approximation of the real value. But the other dimensions, such as the length of the foot, present a large error as shown in Table 3.4. Values obtained in the modified algorithm are closer to the original dimensions. The modified algorithm presents a better-reconstructed image since the surface seems to be smoother, and the approximation to the original value is closer. The disadvantage that the algorithm presents is the running time. The main reason is the fact that in the algorithm, the image processing is considered at the moment of time calculation. Compared with the other two methods, calculation time is increased by three times. Another problem is that the shape of the object only with the bright parts is well recovered. Dark parts are not recovered at all. For these reasons and the fact that the light has to be posed in a certain point to obtain a better result, this algorithm is not suitable for the objectives of the research.

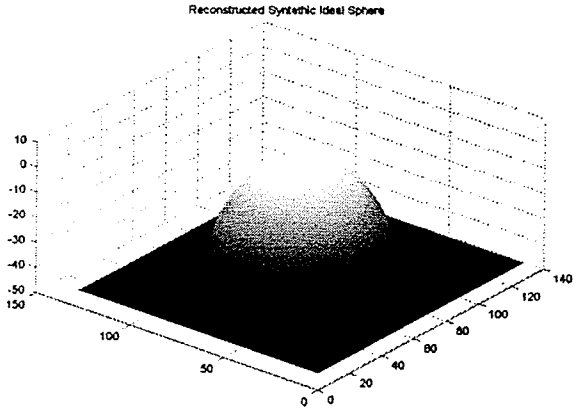




Pentland Method

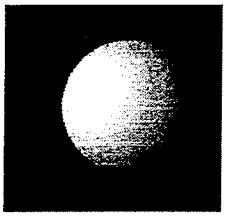


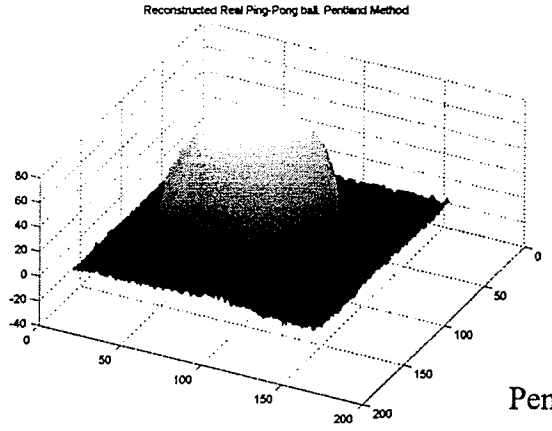
Tsai and Shah Method



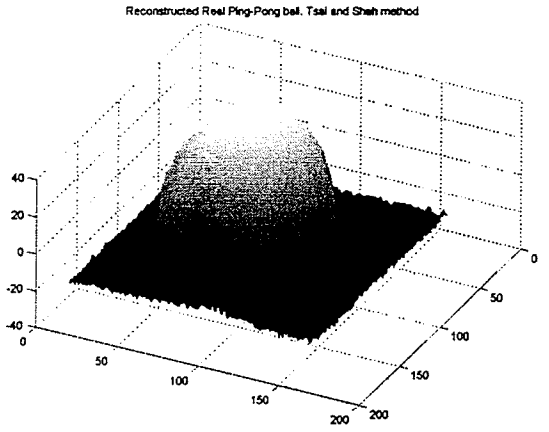
Implemented Method

Fig. 3.18 Synthetic Image reconstructed using different methods

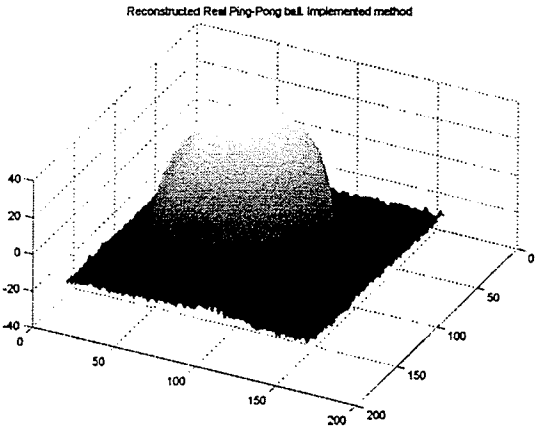




Pentland Method



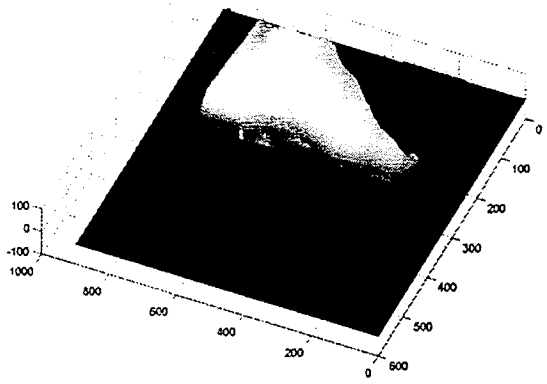
Tsai and Shah Method



Implemented Method

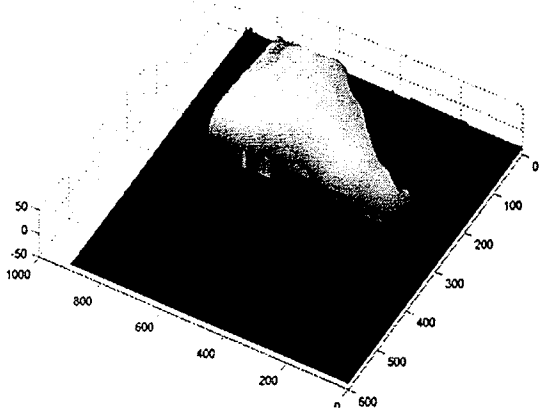
Fig. 3.19 Ping Pong Image reconstructed using different methods

Reconstructed Real Foot, Pentland Method



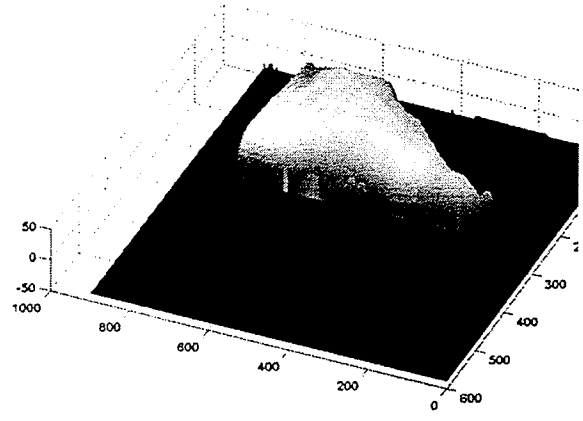
Pentland Method

Reconstructed Real Foot, Tsai and Shah



Tsai and Shah Method

Reconstructed Real Foot.



Implemented Method

Fig. 3.20 Foot Image using different methods

Table 3.1: Comparing the results of the synthetic ball.

Method	Min Value (mm)	Max Value (mm)	Total Difference (mm)	Difference (mm)	Error (%)	Processing Time (sec)
Original Value	0	40	40	0	-	0
Pentland	-10.59	42.32	31.73	9.27	20.675	2.84
Tsai and Shah	-38.03	1.64	36.3	3.6	9.25	2.51
Implemented	-43.85	1.88	41.12	1.12	2.8	4.78

Table 3.2 Comparing the results of the ping-pong ball.

Method	Min Value (mm)	Max Value (mm)	Total Difference (mm)	Difference (mm)	Error (%)	Processing Time (sec)
Original Value	0	40	40	0	-	0
Pentland	-17.25	48.81	31.56	8.44	21.1	14.11
Tsai and Shah	-38.08	8.2	29.88	10.12	25.3	15.99
Implemented	-39.12	6.66	32.46	7.54	18.85	29.4

Table 3.3 Comparing the results of the foot (High Values-Z).

Method	Min Value (mm)	Max Value (mm)	Total Difference (mm)	Difference (cm)	Error (%)	Processing Time (sec)
Original Value	0	80	80	0	-	0
Pentland	-38.03	33.5	71.53	8.47	10.58	25
Tsai and Shah	-38.03	33.96	65	15	18.75	23
Implemented	-38.03	34.5	72.53	7.47	9.33	80.8

Table 3.4 Comparing the results of the foot (X and Y Values).

Method	Min X (mm)	Max X (mm)	Difference X (mm)	Error X (%)
Original Value	0	260	260	-
Pentland	166	806	640	146.15
Tsai and Shah	154	798	644	147.69
Implemented	154	790	636	144.61

A new method has to be tested to get a better result. A static light is required to avoid the problems presented in the SFS approach. The suggested method is Structured Light. The literature review of this method will be discussed in the next chapter.

Chapter 4. Structured Light

A structured light system is based on the projection of a single pattern or a set of patterns onto the measuring scene, which is imaged by a set of cameras. Structured light is an efficient technique for obtaining 3D scene information of an object by using a specially designed light source to project sheets or beams of light with a known a priori spatial distribution on to the object. [24]

A complete overview of the structured light technique is presented in Fig. 4.1. The general classification is based on the coding strategy applied. The sub-classification is according to the code used for projecting the pattern. The idea of this review is to know the existing methods proposed and then develop a new approach. These techniques are discussed as follows.

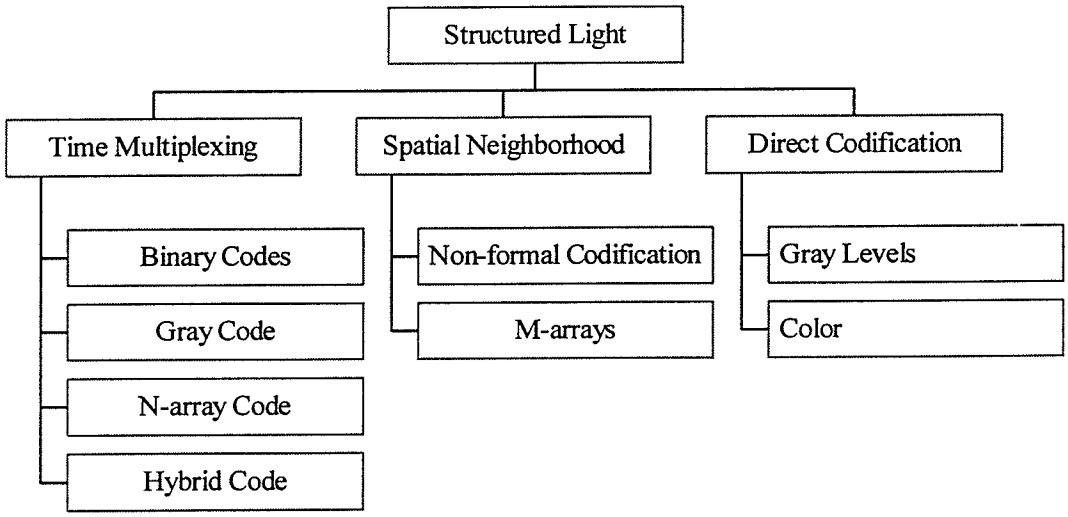


Fig. 4.1 Classification of the Structured Light

4.1 Time-multiplexing strategy

One of the most commonly exploited strategies is based on temporal coding. In this case, sets of patterns are successively projected onto the measuring surface. This technique can achieve high accuracy in the measurements due to two factors. The first one is related to the multiple patterns projected. The location basis tends to be small and therefore a small set of primitives is used, being easily distinguishable from each other. The second factor is due to the position of a pixel. The pixel is encoded more precisely while the patterns are successively projected. As it is shown in Fig. 4.1, according to the techniques used the multiplexing approaches are classified as: a) techniques based on binary codes, where a sequence of binary patterns is used in order to generate binary strips; b) techniques based on n-array codes, where a basis of n primitives is used to generate the projected strips; c) gray code combined with phase shifting, where the same pattern is projected several times, changing the direction in order to increase the number of strips; and d) hybrid techniques, which are a combination of time-multiplexing and neighborhood strategies.

4.1.1 Techniques based on binary codes

There are only two illumination levels commonly used in binary codes, which are coded as 0 and 1. An important characteristic of this technique is that only one of the axes is encoded. Postdamer et al first proposed the projection of a sequence of m patterns to encode 2^m stripes using a plain binary code as shown in Fig. 4.2 [25]. The number of stripes is increasing by a factor of two at every consecutive pattern. Inokuchi et al. improved the codification scheme by introducing gray code instead of plain binary [26].

Its advantages are that consecutive stripes have a pixel distance between the strips, being more robust against noise. Trobina presented a model of coded light based on gray coded patterns [27]. He demonstrated that by simple binarization of the images, the edges of the stripes can be found. After experimental results, he realized that linear interpolation is more accurate than other order interpolations.

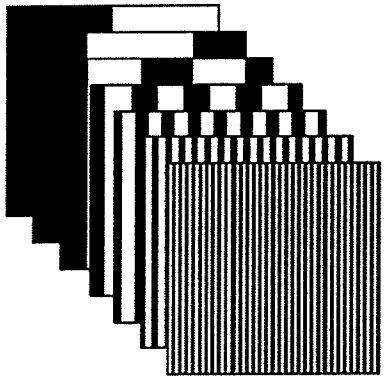


Fig. 4.2 Postdamer Pattern

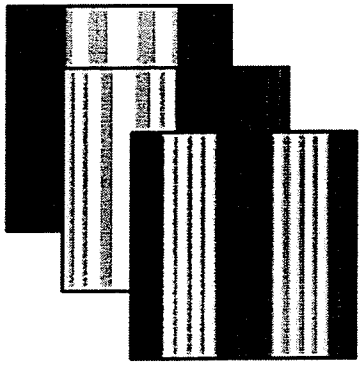


Fig. 4.3 Horn's Pattern

4.1.2 Techniques based on n-array codes

The main drawback of the schemes based on binary codes is the large number of patterns to be projected. However, the fact that only two intensities are projected makes easy for the segmentation of the imaged patterns. There was work concerning the problem of reducing the number of patterns by means of increasing the number of intensity levels used to encode the stripes. The authors that worked in this technique are Horn and Kiryati. They proposed that codes are based on multiple gray levels instead of binary levels as shown in Fig. 4.3 [28]. The aim of the work was to find the smallest set of patterns that meet the accuracy requirements of a certain application producing the best performance under certain noise conditions.

4.1.3 Combination of Gray code and Phase shifting

Patterns based on gray code, as well as binary and n-array codes have the advantage that the codification is made in each pixel. It means no spatial neighborhoods have to be considered. However, the discrete nature of such patterns limits the range resolution. Phase shifting methods exploit higher spatial resolution since they project a periodic intensity pattern several times by shifting it in every projection. The drawback is the periodic nature of the patterns, which introduces ambiguity in the determination of the signal periods in the camera images.

4.1.4 Hybrid methods

In the bibliography, there are some methods based on multiple pattern projection. They use time multiplexing, and also take account of spatial neighborhood information in the decoding process. The idea of Sato consisted of designing a certain binary pattern whose rows have a sharp impulse on its auto-correlation function [29]. The pattern is projected several times by shifting it horizontally several times. For every projection, an image is grabbed, in which the maximum autocorrelation peak of every row is computed. Knowing the phase shift of the corresponding projected pattern, the pixels containing such peaks can be reconstructed by triangulation. Hall-Holt and Rusinkiewicz divided four patterns into a total of 111 vertical stripes that were painted in white and black [30]. Codification is located at the boundaries of each pair of stripes. Fig. 4.4 shows the pattern.

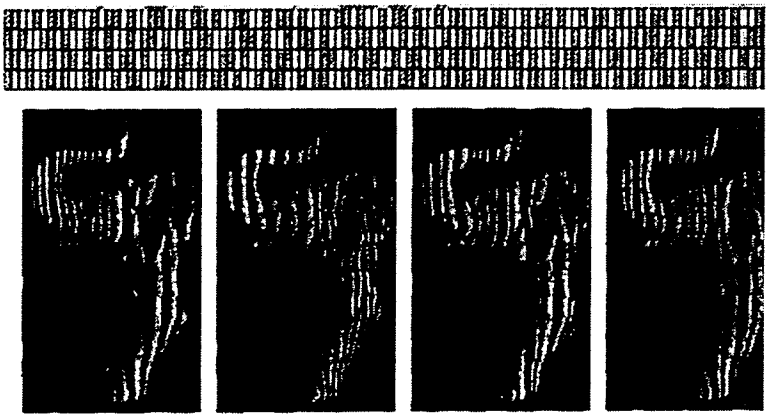


Fig. 4.4 Hal-Holt et al pattern

4.2. Spatial neighborhood

The techniques in this group tend to concentrate the entire coding scheme in a unique pattern. The strip that labels a certain point of the pattern is obtained from a neighborhood of the points around it. However, the decoding stage becomes more difficult since the spatial neighborhood cannot always be recovered and 3D errors can arise. Normally, the visual features gathered in a neighborhood are the intensity or color of the pixels or groups of adjacent pixels included on it. These spatial neighborhood techniques can be classified as: a) strategies based on non-formal codification, where the neighborhoods are generated intuitively; b) strategies based on M-arrays.

4.2.1 Strategies based on non-formal codification

Some authors have proposed techniques based on patterns designed so the required information can be generated with a different label for different regions without using any mathematical coding theory. Maruyama and Abe designed a binary pattern coded with vertical slits containing randomly distributed cuts as shown in Fig. 4.5 [31]. The

system was designed for measuring surfaces with smooth depth changes. The random cuts generate a set of linear segments so that its own length and the lengths of the six adjacent segments determine the position of a segment in the pattern.

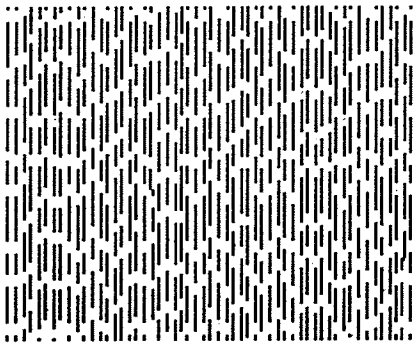


Fig. 4.5. Maruyama's pattern

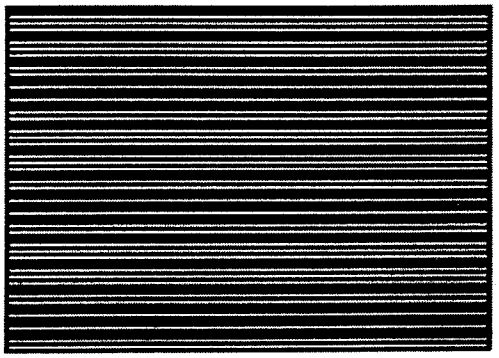


Fig. 4.6. Durdle's pattern

Durdle et al. proposed a periodic pattern composed by horizontal slits encoded with three gray levels [32]. The pattern is formed by the sequence of a band of four black pixels, followed by a band of four white pixels and continued by a band of four half bright pixels. This sequence is repeated in the pattern until it covers all the vertical resolution. An example of pattern codification is shown in Fig. 4.6.

4.2.2 Strategies based on M-arrays

There is a set of authors who have adopted the theory of perfect maps in order to encode a unique pattern. Using M-arrays to codify a pattern means that a bi-dimensional coding scheme is being used. Because every point of the pattern has a different label, vertical and horizontal coordinates are encoded. However, some authors prefer to project additional patterns in order to ease the segmentation part of the system or to carry out an intensity or color normalization. For example, Morita et al. designed a pattern made by

painting black dots on a white background, for the array elements corresponding to symbol 1, as it is shown in Fig. 4.7 [33]. Two patterns are projected onto the surface: the first one contains all the possible black dots in order to locate their centers in the camera image. The second pattern is the M-array representation.

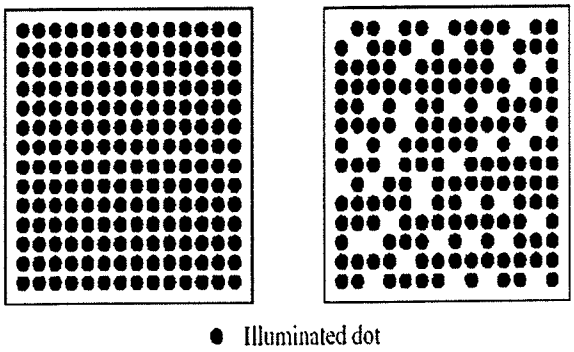


Fig. 4.7 Morita's Pattern

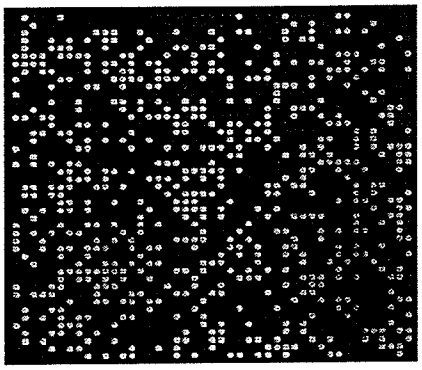


Fig. 4.8 Morano's Pattern

Morano et al. proposed an algorithm for constructing an M-array, fixing the length of stripe, the window property size, the dimensions of the array and a distance between every window [34]. Since every dot is contained in nine windows, the authors applied a voting algorithm where every window proposes a codeword of length. This codeword indicates its position in the pattern for every one of its elements. Then, every imaged dot has up to nine labels proposed by every window that it belongs to. The label with maximum number of votes is the more reliable, so it is used to label the dot. The pattern is shown in Fig. 4.8.

4.3 Direct codification

Direct codification is usually constrained to neutral color objects or pale objects. For this reason, it is necessary to perceive and identify the whole spectrum of colors. Two groups

of methods using direct codification are identified. The first one is known as codification based on gray levels, where a spectrum of gray levels is used to encode the points of the pattern. The other one is the codification based on color, where the advantages of a large spectrum of colors are applied.

4.3.1 Codification based on gray levels

Carrihill and Hummel developed a system called intensity ratio depth sensor. It consists of a linear wedge spread along vertical columns containing a scale of gray levels as shown in Fig. 4.9 [35]. A ratio is calculated between every pixel of the imaged wedge and the same pixel value under constant illumination. Since two patterns must be projected, dynamic scenes are not considered. Miyasaka et al. reproduced the intensity ratio depth sensor by using an LCD projector and a CCD camera. [36]

4.3.2 Codification based on color

The methods belonging to this group use the same principle as the ones discussed in last section by Carrihill and Hummel and Misayaka. However, color is used to encode pixels instead of using gray levels. For instance, Tajima and Iwakawa presented the rainbow pattern as shown in Fig. 4.1.0 [37]. In this method, large sets of vertical narrow slits were encoded with different wavelengths. A large sampling of the spectrum from red to blue was projected. Two images of the scene were taken through two different color filters. By calculating the ratio between both images an index for every pixel is obtained that does not depend on illumination, or on the scene color. Geng improved this approach by using single image of the measuring surface [38].

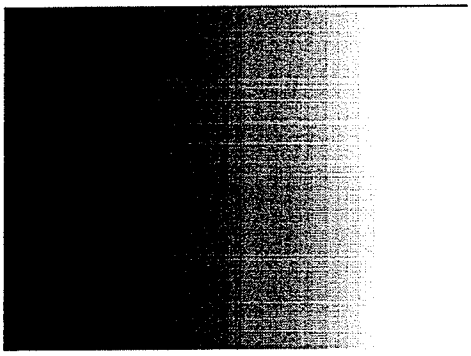


Fig. 4.9 Hummel Pattern

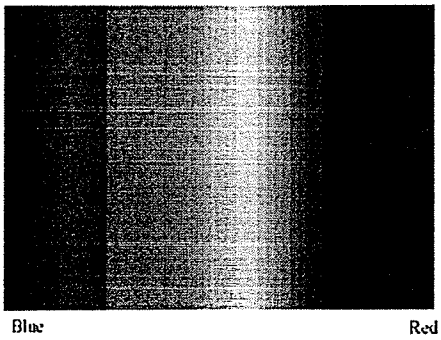


Fig. 4.10 Tajima pattern

4.4. Proposed method of grid projection

In the literature review it was observed that by projecting strips or beams of light with a known spatial distribution on to the object the correspondence problem was solved. It was also reviewed that only the vertical or horizontal axis is used for the reconstruction process. It was mentioned that using a simple binarization of the image the strips of the projected pattern could be found when only black and white strips are projected. Finally, it was demonstrated that only one image was enough for the reconstruction of the object. Following these ideas, a new method is proposed and implemented.

In this research a static mesh with non-colored grids is projected to the object. The image captured by a digital camera is used for the reconstruction. The grid composed of a single projection of $n \times n$ slits is shown in Fig. 4.11. The horizontal and vertical lines have known dimensions and distance between them due to a prior construction process. Moreover, the location of each corner formed by the lines is also known since a mark is presented in the middle of the grid to establish a reference point.

The technique used for unique identification is based on the position of the corners formed by the intersection of the lines. Only the lines are considered and not the squares formed by the intersection of them. Each position of a line can be tracked since the distance between lines is known. When the line is deformed due to the characteristics of the object a comparison between the pixels around the line is performed. If the value is grater than an established distance, the point is considered not part of the same line.

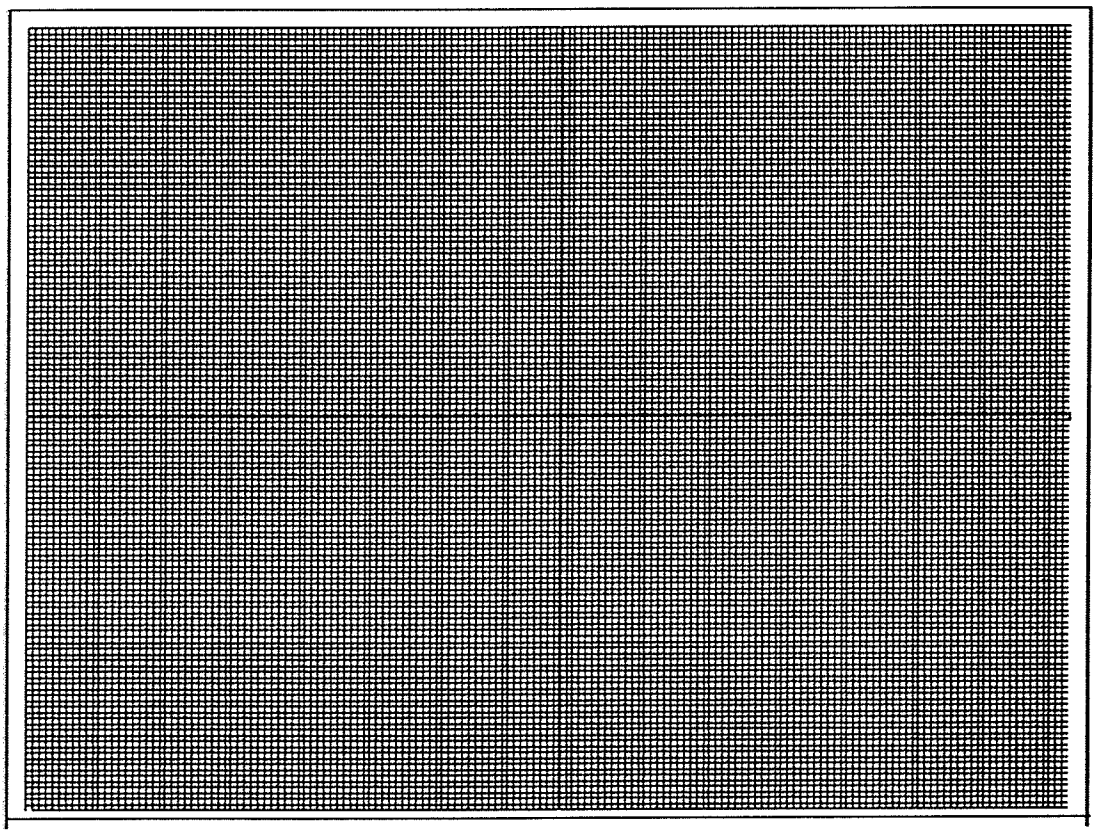


Fig. 4.11 Proposed grid

The slits are designed and projected in such a way that each slit is easily identified and referred to the center of the pattern. Vertical slits are in the X-axis and the horizontal ones are in the Y-axis. This allows finding a slit on a single axis, and finding each crossing point in both axes. Each imaged crossing point can be located, obtaining the (X,Y)

position from where it comes from, and the imaged slit segments can be located knowing where it is vertical or horizontal.

The pattern projection system can be placed in any orientation with respect to the camera because a relation between the projector and the camera is established in the calibration step. When the pattern projection system is used to capture 3D information of an object, a camera takes a picture of a grid projected onto the object under study. The deformed pattern along the image is reconstructed to 3D points by triangulation. The principle follows the stereo approach with the difference that the second camera is substituted by the projector. Using the projector, for each point captured in the picture by the camera, the related coordinates of the grid are known solving the correspondence problem. As the camera, the projector must be calibrated. The values of the projector are estimated in the same way as the camera. It is assumed that the projector works like the second camera.

This approach is intended to obtain a much better approximation of the object shape compared with the SFS method reviewed in Chapter 3. Projecting the grid along the object with minimal spaces between the strips of the grid is assumed to generate an accurate reconstruction. The cost involved in the process is low since only a digital camera and a data projector are required. In the next chapter the complete mathematical model of the proposed method is introduced.

Chapter 5. Grid Projected Method

5.1 Description of the method

The basic idea of the approach is to project a non-colored mesh to the object of interest with a data projector. A digital camera captures the scene where the object is present. These deformed lines are the clue for obtaining the 3D data of the object.

The method is divided into two areas as shown in Fig. 5.1: infrastructure and 3D reconstruction process. The infrastructure is related to the physical devices use for the reconstruction such as the camera, the projector and the calibration patterns. The 3D reconstruction is the procedure followed to complete the reconstruction.

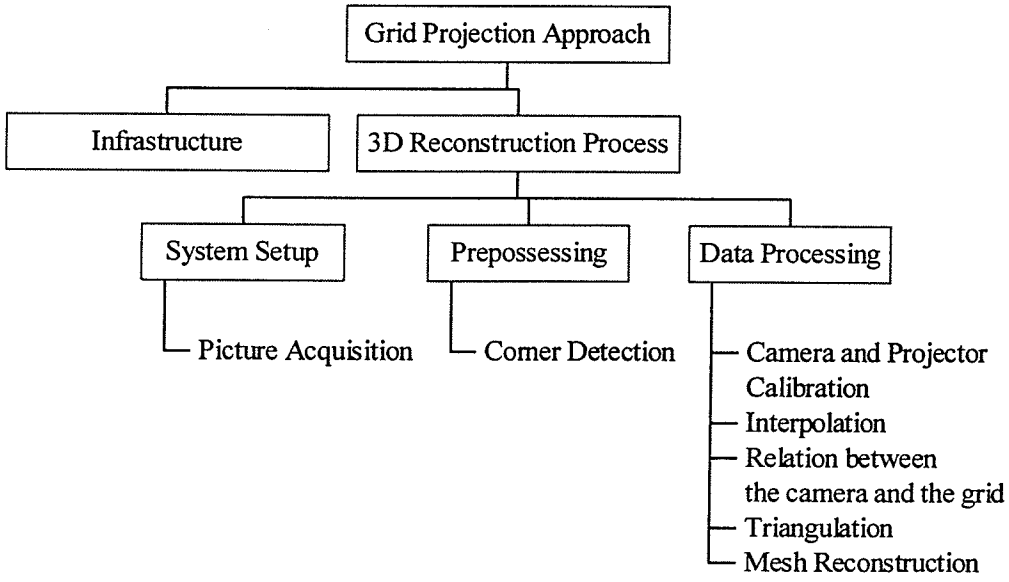


Fig. 5.1 General overview of the method

The overview of the proposed method is as follows. Firstly, the camera and the projector need to be calibrated to obtain the intrinsic and extrinsic values explained in the next

section. A calibration pattern is used to link the image to an established real world coordinate system. This relation is performed by detecting the same points of the calibration patterns in the image coordinate system and the real world coordinate system. Once the instruments are calibrated, they have to be related to a common coordinate system. In this case they are related to the camera's coordinate system. Next the camera takes the scene where the grids are projected into the object under reconstruction. The corners created by the intersection of the lines are deformed in the object. The corners are calculated and related to the center of the projector and the center of the camera. The center of the projector is located in the middle of the projected grid. The center of the picture is the center of the image acquired. The centers are not necessarily in the same position. The triangulation and mesh reconstruction follows the process. The complete skeleton of the method is shown in Fig. 5.2.

The differences of the proposed method from the structured light are: a) the grid projected is static, it is not a set of grids projected in different time; b) the projected grid has only one color, black; c) the thickness of the grid is considerably thinner, only lines are projected and not strips or dots or columns with different color or gray intensity; d) only one picture is taken. The use of the proposed method consists of infrastructure and 3D reconstruction.

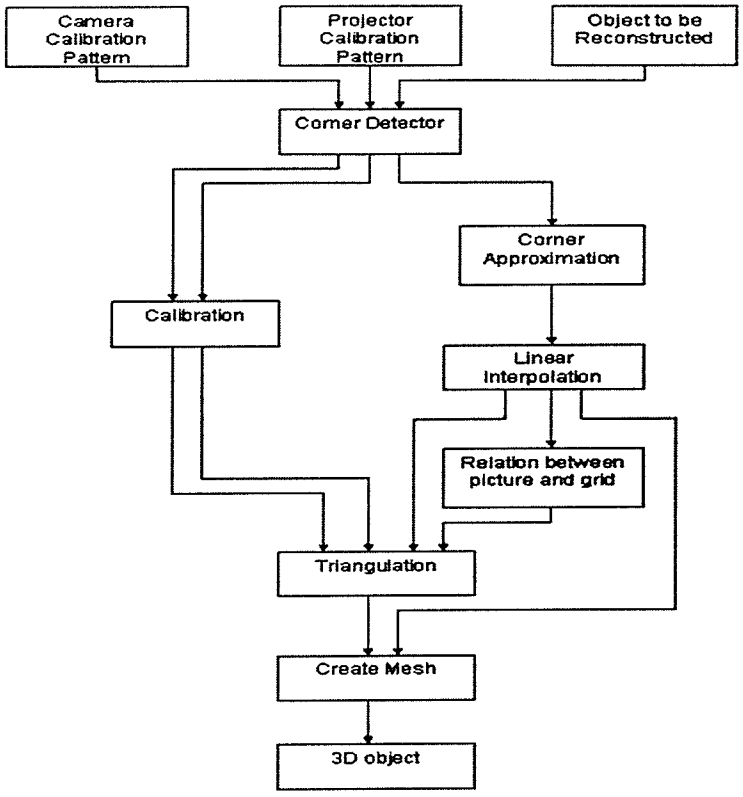


Fig. 5.2 Description of the Data processing steps

5.2 Infrastructure

The basic components of the system include a data projector, a digital camera, a computer, a calibration pattern and the grid projected. Fig. 5.3 shows the basic components of the system.

The projector used is an "EPSON Power-lite 9100" series. The camera is a "Sony FD Mavica" with 2.1Meg Pixels. The resolution used for this research is 1200X1600 pixels. The computer is a Pentium IV, with 1.7GHz. CPU and 520 MB DDR memory.

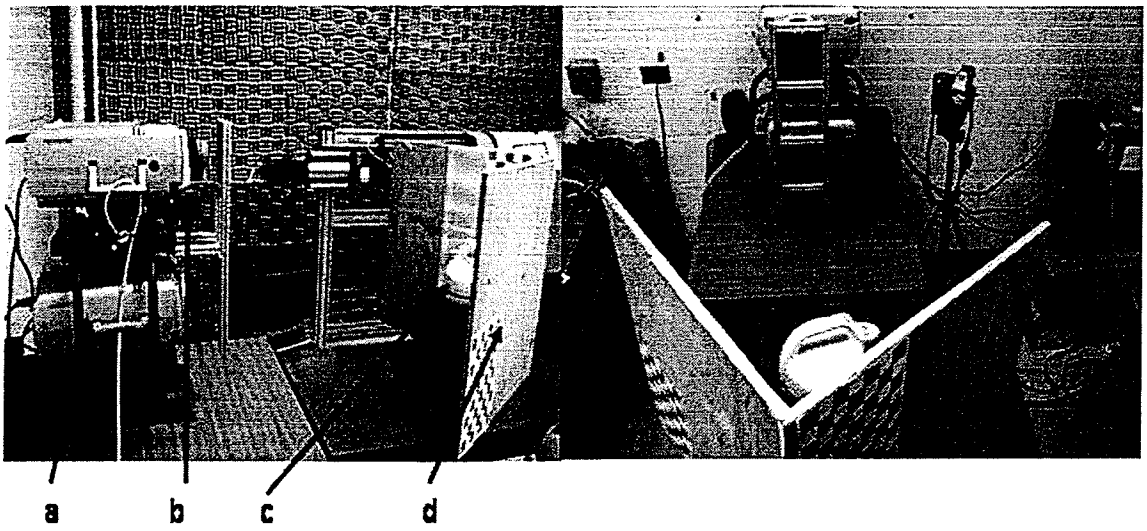
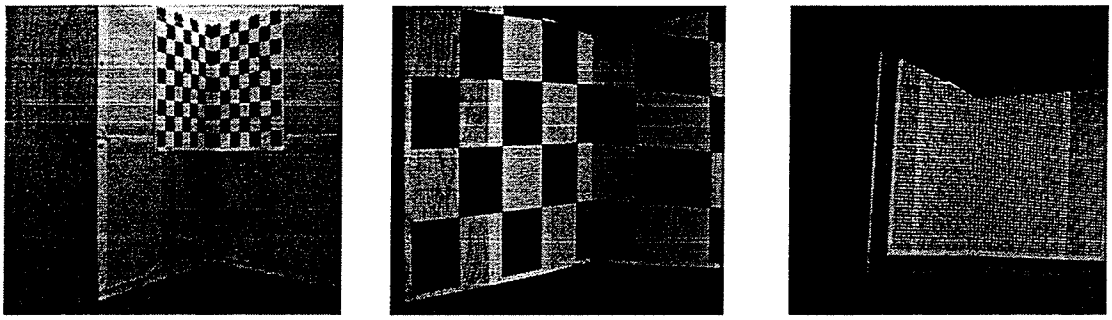


Fig. 5.3 Components of the System, a) projector, b) camera, c) object, d) base

The calibration pattern is formed by equally spaced planar black squares on a white background simulating a chessboard. The pattern is located in a base formed by two perpendicular walls. This system represents the world coordinates: X_w, Y_w, Z_w. Fig. 5.4(a,b) shows the world arrangement of this system as well as the camera and projector calibrations patterns used.

The grid is composed by a single projection of $n \times n$ slits as shown in Fig. 5.4c. Non-colored strips are presented in both vertical and horizontal directions. A mark is presented in the middle of the grid to establish a reference point. The camera and the projector are placed in parallel. The camera is located at the right or left side of the projector.



a) b) c)

Fig. 5.4 World arrangement and Calibration patterns
a) Camera Calibration Pattern, b) Projector Calibration Pattern, c) Grid Projected

5.3 Reconstruction Process

The reconstruction process is divided into three steps as shown in Fig. 5.1. The first step is the system setup that involves the acquisition of the pictures required for the reconstruction. These pictures are used for the camera and projector calibration, as well as the projection of the grid on the scene to be reconstructed. These pictures are shown in Fig. 5.4(a-c).

The next step is preprocessing. The acquired images are prepared to be used in the following step. The last step is data processing for the shape recovery by calculations. The first calculation is the camera and projector calibration. This calibration is performed once only if the camera and the projector are not moved when taking the picture of the projected grid. Then numbers of corners in the object are increased by interpolation. Triangulation is required next for preparing the data for the final step called mesh generation.

5.3.1 System Setup

The first step for the reconstruction is the calibration of the camera and the projector. A physical pattern is used for the camera calibration as shown in Fig. 5.4a. For the projector calibration, the projector displays the pattern on the world system as demonstrated in Fig. 5.4b. The world dimension of each pattern is measured to relate it with picture coordinates. Once the pictures of the calibration patterns are taken, the projector shows the grid. This projected grid, created in MS Excel™, is aligned with the center point of the projector to ensure correspondence. The projected grid is shown in Fig. 5.4c. After this alignment, the object is placed and a single picture is taken and loaded into the computer for the next step.

5.3.2 **Preprocessing**

This step is the preparation of the pictures for further processing. The main goal is to de-noise the pictures and to find the corners presented on them. A corner detector is used to achieve the task.

a) **Corner Detection.**

The detection of feature points in images is essential for many tasks in computer vision, such as structure from motion; object tracking and 3D scene reconstruction from stereo image pairs just to mention some. Gonzales and Woods [39] define corners as connected pixels that lie on the boundary between two regions. He also relates corners as pixel intensity discontinuities in an image that refer to connected chains of edge points.

The corner detector must satisfy the requirements that a) all the corners should be detected; b) no false corners should be detected, c) corner points should be well localized and finally d) the detection should be efficient. Corner detection is typically a three-step process: noise smoothing, enhancement and localization. Noise smoothing is the suppression of the image noise as much as possible, without destroying the true edges. Enhancement is the design of a filter responding to the corners. The filter's output is large at corner pixels and low in other regions of the image, so that corners can be located as the local maxima in the filter's output. Finally the corner localization is decided by which local maxima in the filter's output are corners and which are not. A threshold is normally applied.

The corner detector used is based on the Harris and Stephens algorithm. Harris described what is known as the "Plessey" feature point detector [40]. It estimates the autocorrelation from the first-order derivatives of the image, where at each pixel location a $2X2$ autocorrelation matrix is computed as:

$$A = w * \begin{bmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \left(\frac{\partial I}{\partial x}\right)\left(\frac{\partial I}{\partial y}\right) \\ \left(\frac{\partial I}{\partial x}\right)\left(\frac{\partial I}{\partial y}\right) & \left(\frac{\partial I}{\partial y}\right)^2 \end{bmatrix} \tag{5.1}$$

$I(x,y)$ is the gray level intensity and w is a Gaussian smoothing mask. The filter is used to avoid false corners located due to the image noise. The matrix characterizes the structure of the grey levels. These levels are represented by the eigenvalues of A and

their geometric interpretation. A is symmetric and can be diagonalized by a rotation of the coordinate axes. Thus A as a diagonal matrix is represented by:

$$A = w^* \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}. \tag{5.2}$$

The two eigenvalues, λ_1 and λ_2 are both nonnegative as is explained next. If the neighbour is perfectly uniform, the image gradient vanishes everywhere so A becomes a null matrix and the relation $\lambda_1 = \lambda_2 = 0$ is present. If the neighbours are ideal black and white step edge, the relation $\lambda_2 = 0, \lambda_1 > 0$ and the eigenvector associated with λ_1 is parallel to the image gradient. Finally, if the neighbourhood contains the corner of a black square against a white background, the relation $\lambda_1 \geq \lambda_2 \geq 0$ is expected. The larger the eigenvalues, the stronger their corresponding lines.

A corner is identified by two strong edges; therefore, as $\lambda_1 \geq \lambda_2$, a corner is a location where the smaller eigenvalues λ_2 is large enough.

The used algorithm can be simplified as follows. The input is formed by an image I and two parameters: the threshold on λ_2, τ and the linear size of a square window with size $2N+1$ pixels.

1. The image gradient is computed over the entire image I
2. For each image point p,
    a. The matrix C is formed over a $(2N+1)\times(2N+1)$ neighbourhood Q of p.
    b. λ_2 is computed, getting the smallest eigenvalues of C
    c. if $\lambda_1 \geq \tau$ the coordinates of p are saved into a list L
3. The list L is sorted in decreasing order of λ_2.
4. For the sorted list from top to bottom: for each point p, all points appearing further on in the list that belong to the neighbourhood of p are deleted.

The output is a list of feature points which $\lambda_2 \geq \tau$ and whose neighbourhoods do not overlap. The algorithm `corner.m` is presented in Appendix D.

The results corresponding to the corner detector are presented next. The image tested is an electric kettle where a grid is projected onto it as shown in Fig. 5.5.The pictures show different values of the Gaussian smoothing mask, variable defined as w in equation 5.1, as well as the threshold applied to the images. The values tested were based on the recommended by the authors where the algorithm comes. This is a smoothing constant equal to 2 and thresholding, that is optional, equals to 1000. The corners detected under different conditions are presented in the Figs. 5.6-5.8.

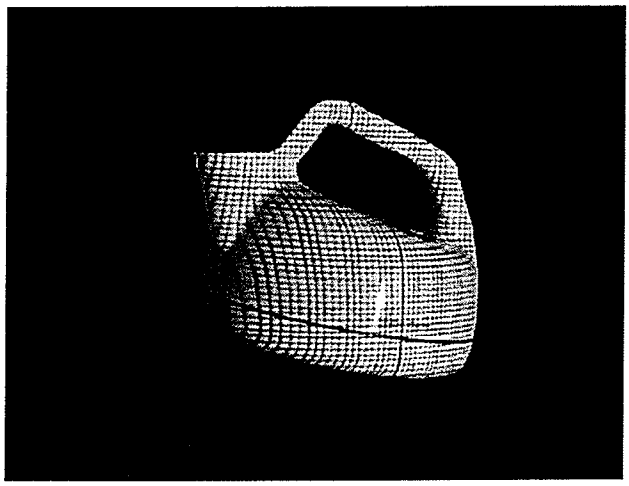


Fig. 5.5 Original image to be tested with different conditions of corner detection

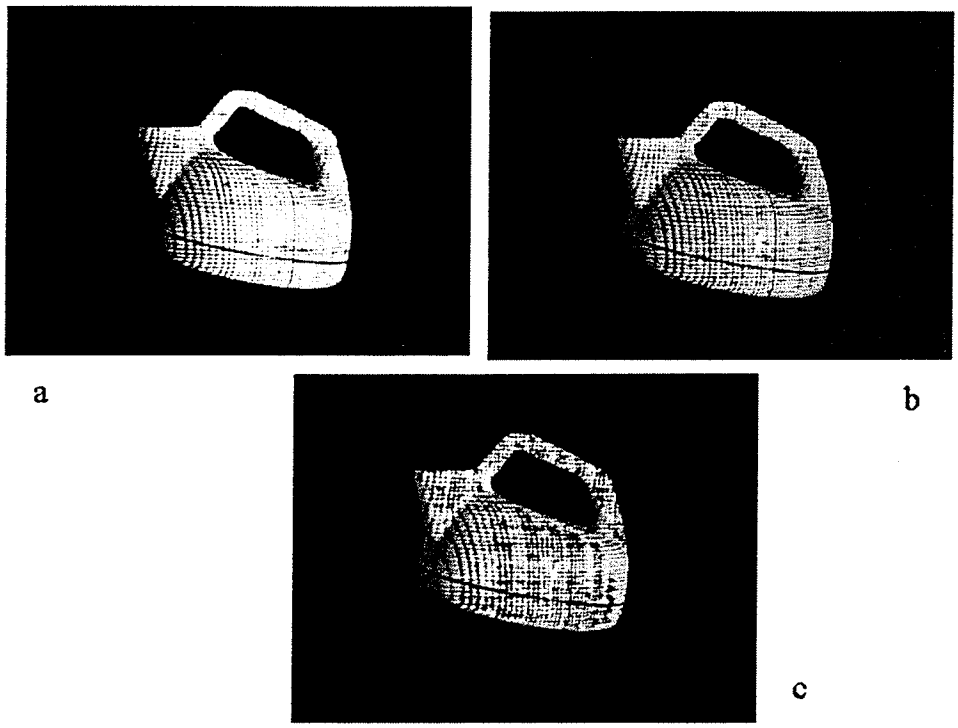


Fig. 5.6 Corners detected with threshold equal to 2500 and a) w=5 b) w=3, c) w=1

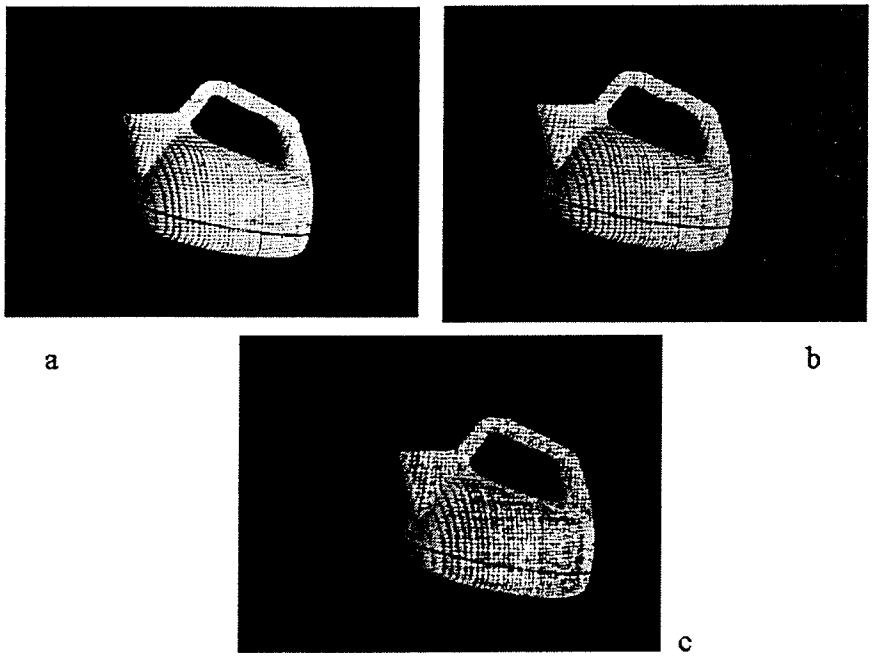


Fig. 5.7 Corners detected with threshold equal to 1000 and a) w=5 b) w =3, c) w =1

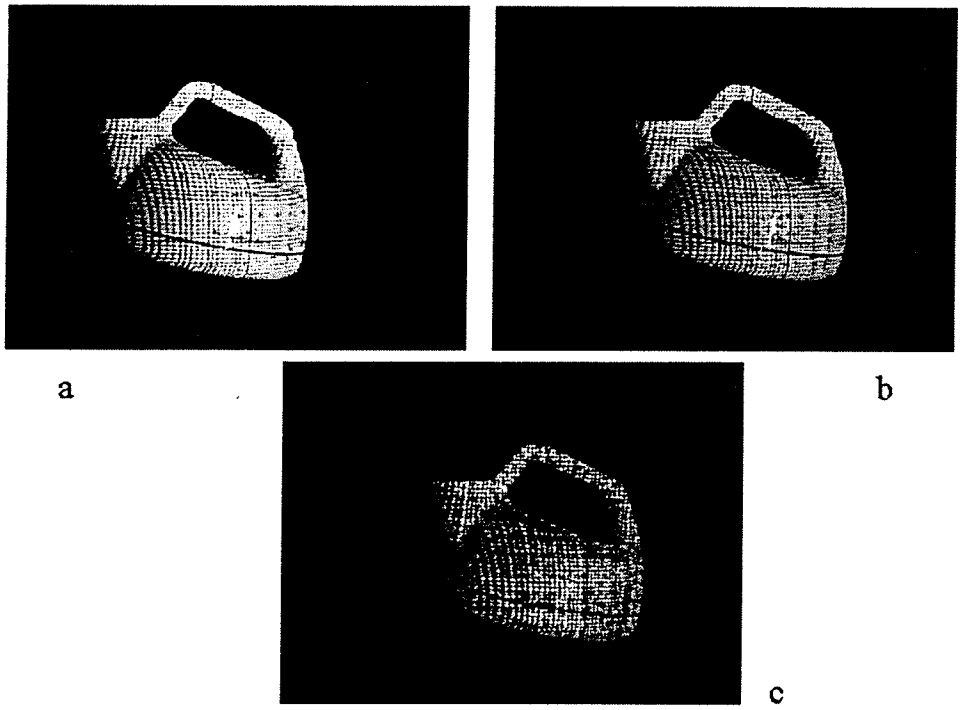


Fig. 5.8 Corners detected with threshold equal to 10 and a) w =5 b) w =3, c) w =1

As it is observed, the difference of the corners detected by varying the smooth constant as well as the threshold in the algorithm allow to acquire different corners even of the same image. Setting the threshold low as well as the smooth constant, bright objects are identified, obtaining a better result.

5.3.3 Data processing

Data processing involved several different operations such as the camera calibration, linear interpolation, relation of the picture and the camera with the center of the grid and finally the triangulation.

a) Camera and Projector Calibration

The goal of the calibration method is to obtain the intrinsic and extrinsic parameters of the camera model. Given the correspondences between a set of point features in the world (X_W, Y_W, Z_W) and the projection in an image (x_{im}, y_{im}), the calibration can be calculated.

In this section the calibration algorithm used is presented. The algorithm is programmed in Matlab. It is assumed that the world reference frame is known. In this case, it is located in the middle of the projected pattern as shown in Fig. 5.4c. It is also necessary to have the pattern's corners in world coordinates (X, Y, Z) as well as the corresponding value in the image (x, y). These values are obtained with the corner detector.

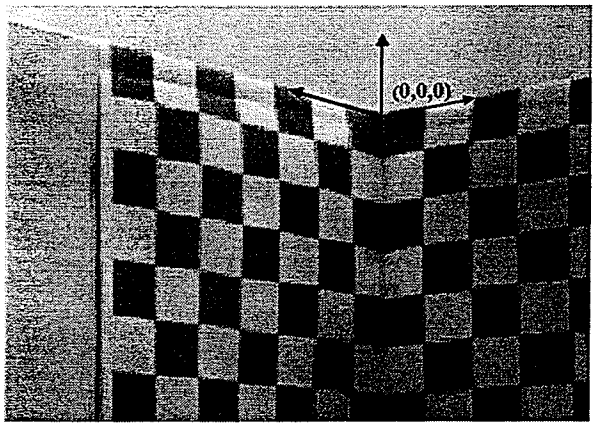


Fig. 5.9 World Reference Frame

The algorithm is used for both, camera calibration and projector calibration. The only difference is related to the calibration pattern used. For the camera a physical pattern is photographed. For the projector, even the calibration pictures are taken with the same camera; the projector projects the pattern. Raskar and Beardsley considered the projector a dual of a camera so that the image projection process can be expressed using the standard pinhole camera model [41]. The internal and external parameters of a projector

can be expressed in the same way as those of the camera. The projector cannot view the illuminated surface, thus, to calibrate a projector is necessary to give the correspondence between six or more 2D projector pixels and corresponding 3D points on a known target object as mentioned by Faugeras [42]. Since the projector is supposed to be a second camera, this calibration is possible.

The extrinsic parameters model the position and orientation of the camera frame with respect to the world reference frame [43]. They define the relation between the camera coordinate system located on the image plane with respect to the world coordinate system located at any point in the scene. These parameters are necessary to link the pixel coordinates of an image point with the corresponding coordinates in the camera reference frame.

The transformation uses a 3D translation vector T and a rotation matrix R. T describes the relative positions of the origins of the two reference frames and R gives the orientation of the coordinate axis of the camera with respect to the world coordinate system. T is represented by

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{5.3}$$

The matrix R is the product of the rotation of ω, ϕ, κ (pitch, roll, jaw) degrees around the axes x, y, z respectively. Each rotation is expressed by the following equations.

$$R_x(\omega)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\omega & -\sin\omega\\ 0 & \sin\omega & \cos\omega\end{bmatrix};\ R_y(\phi)=\begin{bmatrix}\cos\phi & 0 & \sin\phi\\ 0 & 1 & 1\\ -\sin\phi & 0 & \cos\phi\end{bmatrix};R_z(\kappa)=\begin{bmatrix}\cos\kappa & -\sin\kappa & 0\\ \sin\kappa & \cos\kappa & 0\\ 0 & 0 & 1\end{bmatrix} \quad (5.4)$$

So that R is represented as:

$$R=\begin{bmatrix}\cos\phi\cos\kappa & -\cos\omega\sin\kappa+\sin\omega\sin\phi\cos\kappa & \sin\omega\sin\kappa+\cos\omega\sin\phi\cos\kappa\\ \sin\kappa\cos\kappa & \cos\omega\cos\kappa+\sin\omega\sin\phi\sin\kappa & -\sin\omega\cos\kappa+\cos\omega\sin\phi\sin\kappa\\ -\sin\phi & \sin\omega\cos\phi & \cos\omega\cos\phi\end{bmatrix} \quad (5.5)$$

To simplify the representation of the R matrix, it is represented with respect to its three-orientation vector r_i

$$R=\begin{pmatrix}r_{11} & r_{12} & r_{13}\\ r_{21} & r_{22} & r_{23}\\ r_{31} & r_{32} & r_{33}\end{pmatrix} \quad (5.6)$$

As shown in Fig. 5.10, the relation between the coordinates of a point P in a world P_w and camera P_c frames is given by

$$P_c=R(P_w-T) \quad (5.7)$$

or by components

$$\begin{bmatrix}X_c\\ Y_c\\ Z_c\end{bmatrix}=R\begin{bmatrix}X_w\\ Y_w\\ Z_w\end{bmatrix}+T \quad (5.8)$$

$$\begin{aligned}X_c&=r_{11}X_w+r_{12}Y_w+r_{13}Z_w+T_x\\ Y_c&=r_{21}X_w+r_{22}Y_w+r_{23}Z_w+T_y\\ Z_c&=r_{31}X_w+r_{32}Y_w+r_{33}Z_w+T_z\end{aligned} \quad (5.9)$$

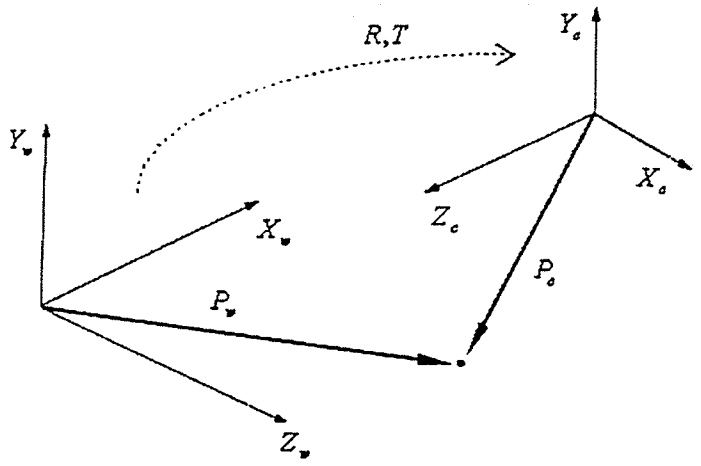


Fig. 5.10 The relation between camera and world coordinate frames

The intrinsic parameters are needed to characterize the optical, geometric and digital characteristics of the viewing camera. They relate a 3D object point with respect to its observable 2D projection on the image plane. The 3D object point must be expressed with respect to the camera coordinate system, and the 2D image point with respect to the image coordinate system. The camera intrinsic parameters are defined as the focal length f, the location of the image centre in pixel coordinates (o_x, o_y), the effective pixel size in the horizontal and vertical direction s_x, s_y, and the radial distortion coefficient k.

The modeling of the intrinsic parameters is divided by [44] in four variables. i) Ideal projection that models the projective relationship between a 3D object points in its 2D image point. ii) Lens distortion that takes into account the discrepancy between the ideal 2D point and the observed 2D point. iii) Pixel adjustment that transforms the 2D observed point from metric coordinates to pixels. And iv) principal point that gives the 2D observed point coordinates with respect to the computer coordinate system.

i) Ideal projection

Consider that any optical sensor can be modeled as a pinhole camera. It is assumed that the camera coordinate system has been located at the optical centre ***C*** with the Z_C axis along the optical axis of the image sensor. Also it is pre-established that the image plane is located at a distance f from the optical centre ***C***, and it is parallel to the plane defined by the coordinate axis X_C, Y_C. From these assumptions, an ideal projection P_U on the image plane must lie on the line passing through the object point ***P*** and the optical centre ***C***. In order to obtain the equations that relate the 3D point P_U with respect to ***P*** and f, the geometric law of the perspective relation will be used as in [45,46]. The perspective relation is given by the following equations:

$$\frac{X_u}{f} = \frac{P_{Xc}}{P_{Zc}}, \frac{Y_u}{f} = \frac{P_{Yc}}{P_{Zc}} \tag{5.10}$$

$$X_u = f\frac{P_{Xc}}{P_{Zc}}, Y_u = f\frac{P_{Yc}}{P_{Zc}} \tag{5.11}$$

Therefore, a single intrinsic parameter has to be considered in order to model the ideal projection; this parameter is the focal length f.

ii) Lens distortion

Weng et al. [47] demonstrated that as a result of some imperfections in the design and assembly of the lenses, the linear relationship of the perspective projection is not true. Lens distortion can be modeled by a radial and tangential approximation. Radial distortion causes an internal or external displacement of a given image point from its

ideal projection. This type of distortion is mainly caused by imperfect radial curvature of the lens. The displacement given by the radial distortion can be modelled by the following equations

$$\begin{aligned} x &= X_d(1 + k_1 r^2 + k_2 r^4) \\ y &= Y_d(1 + k_1 r^2 + k_2 r^4) \\ r &= \sqrt{X_d^2 + Y_d^2} \end{aligned} \tag{5.12}$$

Where X_d, Y_d are the coordinates of the distorted points, r is the radial distance of the observed projection from the projection of the focal point on the image plane. And $k_1, k_2 \ldots$ are the coefficients of the radial distortion. Since they are usually very small, radial distortion is ignored whenever high accuracy is not required in all regions of the image, or when the peripheral pixels can be discarded.

Tangential distortion models the optical system subjected to various degrees of decentre ring. That is, the optical centres of the lens elements are not strictly collinear. Most of the authors have affirmed that radial distortion is the most important lens distortion.

iii) Pixel adjustment

The pixel adjustment is based on the transformation of the real projection on the retinal plane from metric coordinate to the coordinates of the image plane in pixels. This transformation is based on a simple scale adjustment of both axes shown in equation 5.11. X_d, Y_d are the real projection with lens distortion in metric coordinates and X_p, Y_p are the same point expressed in pixels.

$$X_p = s_x X_d$$
$$Y_p = s_y Y_d \tag{5.13}$$

Fig. 5.1 shows these relations.

iv) Principal Point

The principal point, also known as image centre, is defined as a point o_x, o_y on the image plane given by the intersection of the optical axis of the camera and the image plane. The principal point is the projection of the focal point on the image plane that is given in pixels.

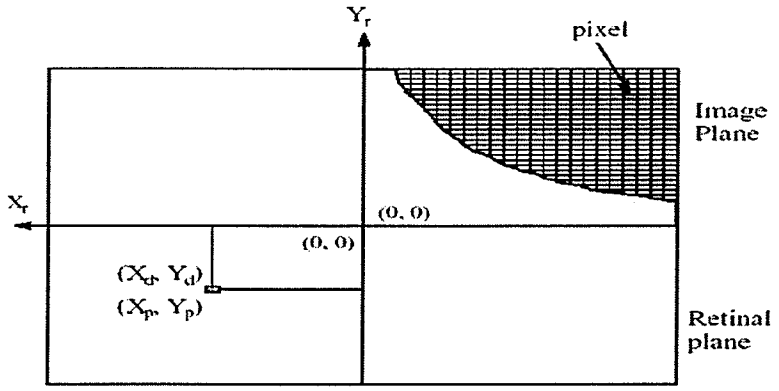


Fig. 5.11 Transformation from metric coordinates of the retinal plane to pixel coordinates of the image plane

The origin of the coordinate system in the computer image plane is located at the left superior corner of the image. Fig. 5.12 illustrates this relation. A translation is required in order to place the principal point from the corner of the image to its real location on the image plane. This translation is given by the values o_x, o_y, with respect to each axe of the coordinate system. o_x and o_y axes of the computer image plane are inverted from the X_r and the Y_r axe of the retinal image plane.

Given a point x_{im}, y_{im} expressed with respect to the camera coordinates system, the same point will be expressed with respect to the computer image coordinate system using the following equations.

$$x_{im} = -(X_p + o_x)s_x$$
$$y_{im} = -(Y_p + o_y)s_y \qquad (5.14)$$

The resulting point x, y is the point observed in pixels on the image plane of the computer.**Error!**

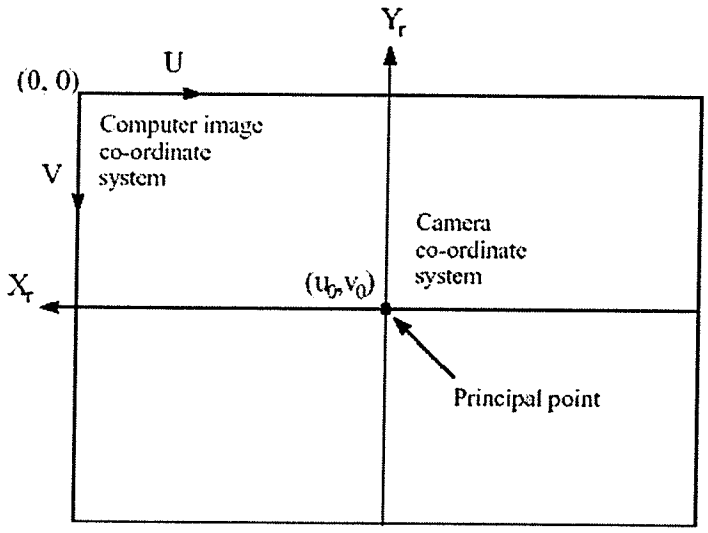


Fig. 5.12 Transformation from the coordinate system of the camera image plane to the coordinate system of the computer image

v) The whole camera model

Now it is possible to establish a relation between the pixel coordinates of an image point with the world coordinates of the corresponding 3D point. The equations (5.7) and (5.14) are the basic equations of the perspective projection in the camera frame to obtain the relation

$$-(x_{im}-o_x)s_x = f\frac{R_1^T(P_w-T)}{R_3^T(P_w-T)}$$
$$-(x_{im}-o_y)s_y = f\frac{R_2^T(P_w-T)}{R_3^T(P_w-T)} \qquad (5.15)$$

where R_i, $i=1,2,3$, is a 3D vector formed by the $i-th$ row of the matrix R. Last equations define the 3D coordinates of a point in the world reference to the image coordinates of the corresponding image point, via the camera extrinsic and extrinsic parameters. If the radial distortion is not considered, the intrinsic parameters and extrinsic parameters are combined as follows:

$$x-o_x = -f\frac{r_{11}(X_w)+r_{12}(Y_w)+r_{13}(Z_w)+T_x}{r_{31}(X_w)+r_{32}(Y_w)+r_{33}(Z_w)+T_z}$$
$$y-o_y = -f\frac{r_{21}(X_w)+r_{22}(Y_w)+r_{23}(Z_w)+T_y}{r_{31}(X_w)+r_{32}(Y_w)+r_{33}(Z_w)+T_z} \qquad (5.16)$$

Also (5.13) can be rewritten in a matrix product of two matrices defined as: M_{int} and M_{ext} as

$$M_{\text{int}} = \begin{pmatrix} -f/s_x & 0 & u_o \\ 0 & -f/s_y & v_o \\ 0 & 0 & 1 \end{pmatrix}; \; M_{ext} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{pmatrix} \qquad (5.17)$$

The last equation is a 3X3 M_{int} only depending on the intrinsic parameters, while the 3X4 matrix M_{ext} only depends on the extrinsic parameters. A forth coordinate is added to obtain a linear matrix equation describing perspective projections. The linear matrix equation of perspective projection is then

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = M_{\text{int}} M_{ext} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{23} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{5.18}$$

The solution is presented based on [41, 42, 47, 48]. For convenience, $M_{\text{int}} M_{ext}$ are substitute for M :

$$\begin{pmatrix} u_i \\ v_i \\ w_i \end{pmatrix} = M \begin{pmatrix} X_i^w \\ Y_i^w \\ Z_i^w \\ 1 \end{pmatrix} \tag{5.19}$$

It is assumed that $o_x = o_y = 0$ and $s_x = s_y = 1$

$$M = \begin{pmatrix} -fr_{11} & -fr_{12} & -fr_{13} & fR_1^T T \\ -fr_{21} & -fr_{22} & -fr_{23} & fR_2^T T \\ r_{31} & r_{32} & r_{33} & fR_3^T T \end{pmatrix} \tag{5.20}$$

and

$$\begin{aligned} x &= \frac{u_i}{w_i} = \frac{m_{11} X_i^w + m_{12} Y_i^w + m_{13} Z_i^w + m_{14}}{m_{31} X_i^w + m_{32} Y_i^w + m_{33} Z_i^w + m_{34}} \\ y &= \frac{u_i}{w_i} = \frac{m_{21} X_i^w + m_{22} Y_i^w + m_{23} Z_i^w + m_{24}}{m_{31} X_i^w + m_{32} Y_i^w + m_{33} Z_i^w + m_{34}} \end{aligned} \tag{5.21}$$

Note that the sub-index of the variables is changed since more than one point is analyzed at the same time.

The matrix M is defined up to an arbitrary scale factor and has therefore only eleven independent entries. The eleven entries are the fact that m_{ij} is divided by m_{11}. At least

$N = 6$ world-image point correspondences to solve for the entries of M are required. The values of m_{ij} are determined through a homogeneous linear system formed by equations 5.21. However, with the use of the calibration pattern many more correspondences and equations can be obtained because more points are located. Therefore, M can be estimated through the least square technique. It is assumed that N matches are given for the homogeneous linear system:

$$Am = 0 \tag{5.22}$$

where

$$A = \begin{pmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1Z_1 & -x_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1Z_1 & -y_1 \\ X_2 & Y_2 & Z_2 & 1 & 0 & 0 & 0 & 0 & -x_2X_2 & -x_2Y_2 & -x_2Z_2 & -x_2 \\ 0 & 0 & 0 & 0 & X_2 & Y_2 & Z_2 & 1 & -y_2X_2 & -y_2Y_2 & -y_2Z_2 & -y_2 \\ . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . & . & . \\ X_N & Y_N & Z_N & 1 & 0 & 0 & 0 & 0 & -x_NX_N & -x_NY_N & -x_NZ_N & -x_N \\ 0 & 0 & 0 & 0 & X_N & Y_N & Z_N & 1 & -y_NX_N & -x_NY_N & -x_NZ_N & -y_N \end{pmatrix} \tag{5.23}$$

and

$$m = [m_{11}, m_{12}, ..., m_{33}, m_{34}]^T \tag{5.24}$$

The vector m can be recovered from SVD related techniques as the column V corresponding to the zero singular value of A, with $A = UDV^T$.

The last column of V is denoted as $\overline{m}$ so the next relations are established

$m = \kappa\overline{m}$ or $\overline{m} = \gamma m$ where κ is the scale factor and $\gamma = \frac{1}{\kappa}$.

Matrix M is calculated with the help of the calibration pattern. Different positions in the world coordinate system are taken, and then these positions are located in the camera coordinate system. The same points in the image taken by the camera can also be found by this relation. The code of this section called `Projmat.m` is presented in Appendix E.

The camera parameters are calculated as functions of the estimated projection matrix M

The full expression for the entries of M is rewritten as follows:

$$M = \begin{bmatrix} -f_x r_{11} + o_x r_{31} & -f_x r_{12} + o_x r_{32} & -f_x r_{13} + o_x r_{33} & -f_x T_x + o_x T_z \\ -f_y r_{21} + o_y r_{31} & -f_y r_{22} + o_y r_{32} & -f_y r_{23} + o_y r_{33} & -f_y T_y + o_y T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \tag{5.25}$$

The vectors are defined as:

$$\begin{aligned} q_1 &= [\hat{m}_{11}, \hat{m}_{12}, \hat{m}_{13}]^T \\ q_2 &= [\hat{m}_{21}, \hat{m}_{22}, \hat{m}_{23}]^T \\ q_3 &= [\hat{m}_{31}, \hat{m}_{32}, \hat{m}_{33}]^T \\ q_{41} &= [\hat{m}_{14}, \hat{m}_{24}, \hat{m}_{34}]^T \end{aligned} \tag{5.26}$$

Since M is defined up to a scalar factor

$$\hat{M} = \gamma M \,. \tag{5.27}$$

The scalar factor is obtained from q_3 that is the last row of the rotation matrix R.

From the last row of M it is established that

$$T_z = \sigma \hat{m}_{34} \tag{5.28}$$

and

$$r_{31} = \sigma \hat{m}_{31} \tag{5.29}$$

where $\sigma = \pm 1$

Taking the dot products of q_3 with q_1 and q_2 it is found that

$$\begin{aligned} o_x &= q_1^T q_3 \\ o_y &= q_2^T q_3 \end{aligned} \tag{5.30}$$

and

$$\begin{aligned} f_x &= \sqrt{q_1^T q_1 - u_o^2} \\ f_y &= \sqrt{q_2^T q_2 - v_o^2} \end{aligned} \tag{5.31}$$

The remaining parameters are obtained as

$$\begin{aligned} r_{1i} &= \sigma(o_x \hat{m}_{31} - \hat{m}_{1i}) / f_x \\ r_{2i} &= \sigma(o_y \hat{m}_{31} - \hat{m}_{2i}) / f_y \\ T_x &= \sigma(o_x T_z - \hat{m}_{14}) / f_x \\ T_y &= \sigma(o_y T_z - \hat{m}_{24}) / f_y \end{aligned} \tag{5.32}$$

The sign of σ is obtained from $T_z = \sigma \hat{m}_{34}$. It is positive if $T_z > 0$ and negative if $T_z < 0$

The Matlab code for this part is called `CameraParameters.m`. The code used for the complete representation of the model is `FinalCalib.m`. The code is presented in Appendix E

b) Linear interpolation

Linear interpolation is a process employed in linear algebra, and numerous applications including computer graphics and numerical analysis. In this research, it is applied to

increment the number of points detected in the corner detection algorithm. A incrimination of points represents a better approach at the moment of reconstruction. Fig. 5.13 shows two points (x_1, y_1) distantied to (x_0, y_0).

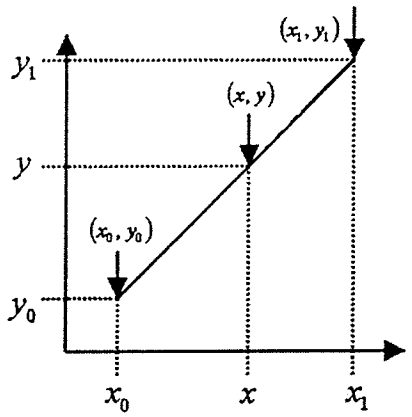


Fig. 5.13 Linear Interpolation

To find an intermediate value (x, y) between these points, by inspecting the figure we see that:

$$\frac{y - y_0}{y_1 - y_0} = \frac{x - x_0}{x_1 - x_0} \tag{5.33}$$

By manipulating this algebraically, and writing

$$\alpha = \frac{x - x_0}{x_1 - x_0} \tag{5.34}$$

It is obtained:

$$y = (1 - \alpha) y_0 + \alpha y_1 \tag{5.35}$$

that is the same as:

$$y = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) \tag{5.36}$$

The same formula can easily be derived for x when y is known. Appendix F shows the Matlab file called `linearinterpol.m`.

c) Relation between camera and projector

Once the corners of the grid of the desired object are detected, they are related to the image coordinate system. This coordinate system is referred to the middle point of the image size. The image used is a 1600X1200 pixels, the centre of the image is located at the point 800X600. A simple algorithm assigns the value corresponding to the image coordinate system. Corners over the middle value of the image are reduced by that value and vice versa.

After corners are translated to image coordinate system, the position corresponding to the lines of the grid can be found. An average value of the corners is calculated and set as the middle point where the value is (0,0). In most of the cases the value corresponds with the value of the centre of the grid previously selected. This is the fact that the centre of the grid is oriented to the centre of the object too. Both algorithms are presented in Appendix G.

d) Triangulation

The simplest case arises when the optical axes of the camera and projector are parallel, and the translation of the projector is only along the X axis. For this case lets consider the optical setting as it is shown in the Fig. 5.14.

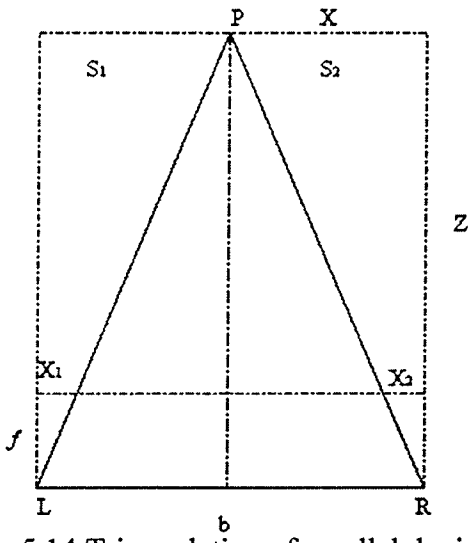


Fig. 5.14 Triangulation of parallel devises

L is a pinhole camera and R is a projector with parallel optical axes referred to left and right respectively. f is the focal length of both devices. The baseline is perpendicular to the optical axes. Let b be the distance between the two lens centers. XZ is the plane where the optical axes lie, XY plane is parallel to the image plane of both devices, X axis equals the baseline and the origin O of (X,Y,Z) world reference system is the lens center of the left camera. S_1 and S_2 are projections of the base line. Tree triangles are identified as

$$\frac{X_1}{S_1} = \frac{f}{Z} = \frac{X_2}{S_2}. \tag{5.38}$$

S_2 is established form the following relation:

$$S_2 = b - S_1 \tag{5.39}$$

By substituting 5.39 in 5.38, the relations obtained are:

$$\frac{X_1}{S_1} = \frac{X_2}{b - S_1}$$
$$\frac{X_1}{S_1} = \frac{X_2}{b - S_1} \qquad (5.40)$$
$$BX_1 = S_1 X_2 + S_1 X_1 = S_1 (X_1 + X_2)$$
$$S_1 = \frac{X_1 b}{X_1 + X_2}$$

Finally the values of X Y and Z are obtained from:

$$Z = \frac{S_1 f}{X_1} = \frac{f}{X_1} \cdot \frac{X_1 b}{X_1 + X_2} = \frac{bf}{X_1 + X_2}$$
$$X = \frac{X_1 Z}{f} = b - \frac{X_2 Z}{f} \qquad (5.41)$$
$$Y = \frac{Y_1 Z}{f}$$

When the devises are not parallel, the camera can be rotated with respect to the projector in the three axes. The relation between triangles is established in the same way as when the camera and projector are parallel.

In the case of rotation around the Y axis (θ) the optical axes are not parallel, but they both lie on the XZ plane, so they intersect in a point $(0,0,Z_0)$, that is called fixation point and could also be behind the camera and projector $(Z_0 < 0)$. If theta is the rotation angle, then

$$Z_0 = \frac{b}{\tan(\theta)} \qquad (5.42)$$

Under small angle approximation, it still assumed the right image plane to be parallel to the left image plane and hence to XY plane. In this case:

$$Z = \frac{b * f}{\frac{x1 - x2 + (f * b)}{Z_0}}$$
$$X = \frac{X1 * Z}{f} \tag{5.43}$$
$$Y = \frac{y1 * Z}{f}$$

Rotation around X axis (ϕ) only affects the Y coordinate in reconstruction. The stereo triangulation is

$$Z = \frac{(b * f)}{x1 - x2}$$
$$X = \frac{x1 * Z}{f} \tag{5.44}$$
$$Y = \frac{y1 * Z}{Z + \tan(\phi) * Z}$$

Rotation around Z axis (φ) is usually dealt with by rotating the image before applying matching and triangulation. In the general case, given the translation vector T and rotation matrix R describing the transformation from left camera to right camera coordinates, the equation to solve for stereo triangulation is

$$p' = RT(p - T) \tag{5.45}$$

where p and p' are the coordinates of P in the left camera and right projector coordinates respectively, and RT is the transpose matrix of R. The triangulation algorithm is presented in Appendix H.

e) Mesh Generation

The mesh generation is the process where the cloud of 3D points is processed to create a surface. The first step is to approximate all the points that are together with an average of their neighbours. Not all the points have a valid number of neighbours. The number of valid neighbours is a parameter that can vary according to points desired to be considered. A good value is four considering the neighbours surrounding the desired pixel. The next step is to smooth the generated mesh. For each point, the number of valid neighbours close to the point in a certain distance is calculated. That value is replaced by the average between the point and the neighbour. The final step is to generate triangles. Four neighbour points at a time are taken. Two triangles using the shortest diagonal are created only if the points are considered close to each other. If there are only three points and if they are close to each other, then they are connected. If there are only two or less valid points, they will not be connected.

The next chapter discusses the results of the camera and projector calibration. The testing and comparison of the developed method with the laser scanner are also presented.

Chapter 6. Experiments and Results

The objective of the experiments is to show and compare the performance of the proposed approach. Three objects are used as samples to study the performance of the method. The evaluation is based on the comparison of the accuracy obtained by the developed system and a commercial laser scanner. The three objects are an electric kettle, a volleyball and a foot model.

In this chapter, following the methodology presented in Fig. 5.2, the calibration and results are presented. Then, the analysis of the reconstructed objects is followed. Finally, the reconstructed objects are compared with the results produced by a single shot of the laser scanner.

6.1 Calibration

The camera and projector need to be calibrated before performing the reconstruction process. The internal and external parameters, as well as the relation between the camera and the projector must be established. To achieve this goal two measurements of each calibration pattern are required to establish the world image relation. The first one is a real measurement of the corners presented in each pattern. The second goal is to find the same points in the image using the corner detector algorithm. The calibration algorithm is performed to obtain the intrinsic and extrinsic parameters of the camera and projector.

Fig. 6.1 and Fig. 6.2 show the calibration pattern and the corner detected for the camera and the projector respectively. The characteristics of the calibration pattern are discussed in section 5.2. Table 6.1 presents the result of the camera and projector calibration.

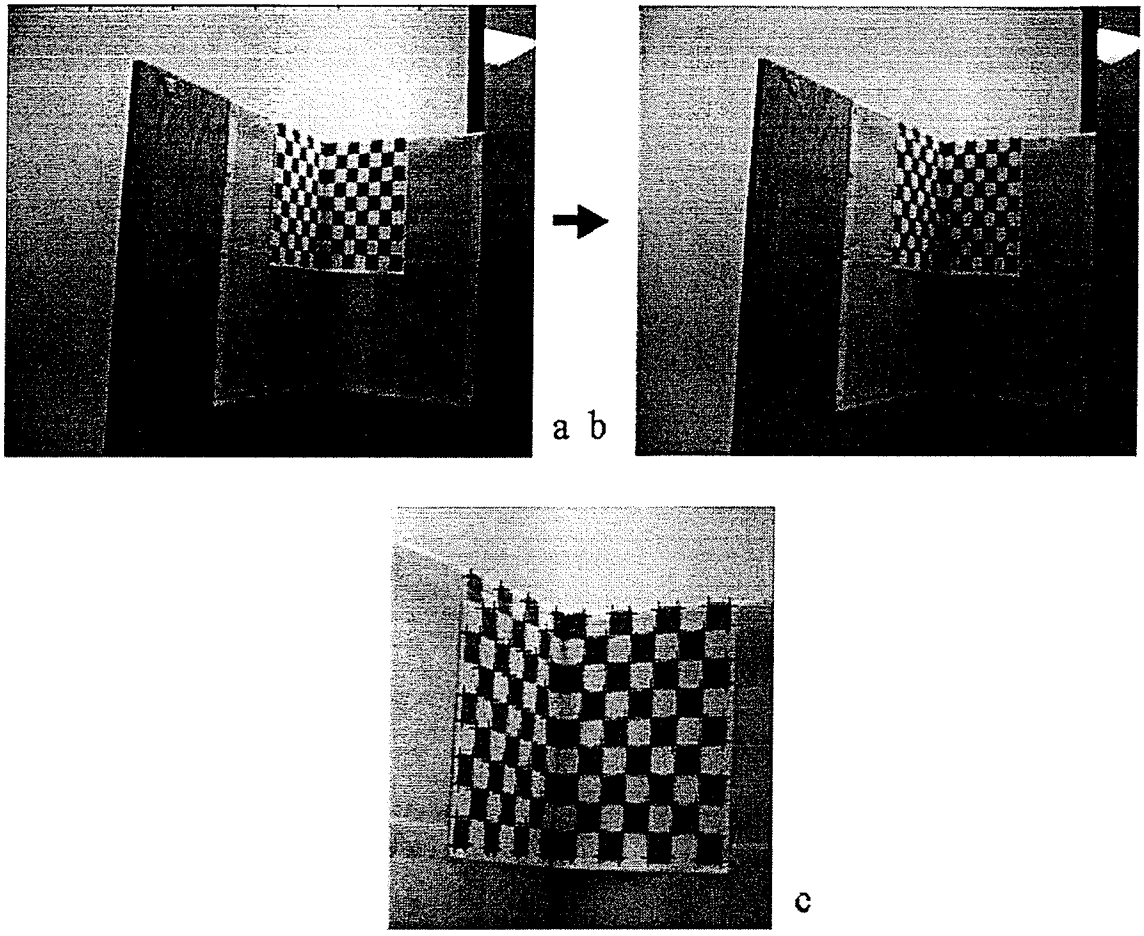
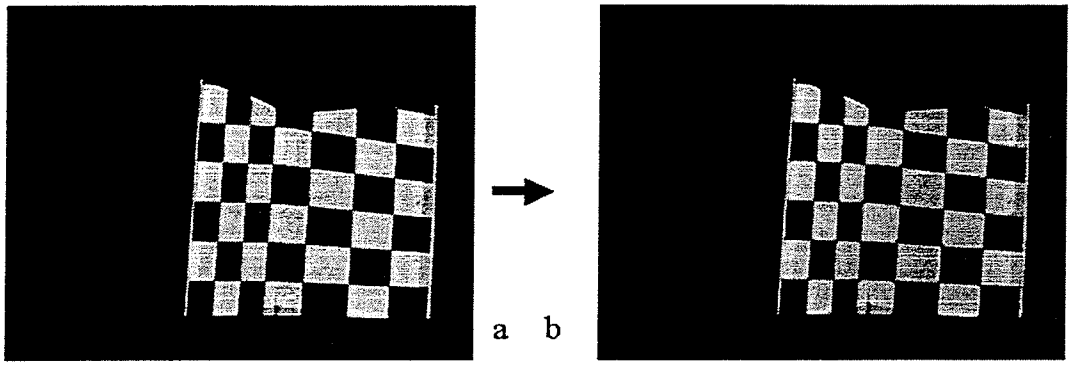


Fig. 6.1 Camera Calibration Results a) Calibration Pattern, b) Corner detected, c) Zoom of the corner detected.



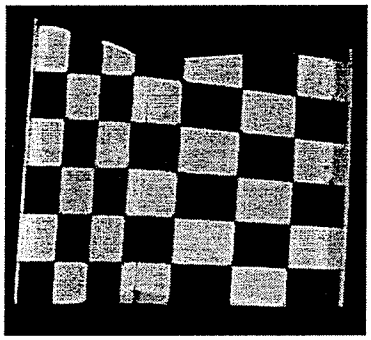


Fig. 6.2 Projector Calibration Results a) Calibration Pattern, b) Corner detected, c) Zoom of the corner detected.

Table 6.1 Calibration Results

Camera Calibration						
R=	0.941	-0.337	0.017	T_x = -7.021	f_x = 1283.4	o_x = 410.79
	-0.037	-0.115	-0.987	T_y = 11.867	f_y = 1046.5	o_y = 128.62
	0.336	0.928	-0.158	T_z = -99.263		
Projector Calibration						
R=	0.292	-0.956	0.003	T_x = 4.657	f_y=581.82	o_x=505.55
	-0.022	-0.011	-0.999	T_y = 12.906	f_y=604.32	o_y=166.26
	-0.956	-0.292	0.024	T_z = -119.61		

The results obtained are similar to different values calculated by Bouget [49] where different methods are tested, therefore the calibration is considered valid.

6.2 Reconstructed objects

Three objects were tested by projecting meshes with different distance between lines. One mesh has *12X12* pixel separation between lines in both directions x and y. Another one was a thinner mesh where the distance between lines was *6X6* pixels. Finally a *3X3* pixel separation between lines was also projected. It was noticed that long distances between grids were not suitable for the reconstruction process. Big holes were found in the surface making the object not suitable for interpolation. Smaller distances between grids were not supported either because the space between lines was not distinguished on the captured image. The mesh that generates better results was the *6X6* pixels spaced mesh. It is difference from the other meshes tested, this mesh is well identified by the camera, the corners can be detected easily and the interpolation can be achieved due to the short distance between the grids. The image resolution used for the acquisition was 1200X1600 pixels.

The first object tested using this mesh was an electric kettle. The projected mesh onto the object is presented in Fig. 6.3 (a). The corner detection algorithm was performed generating good results as shown in Fig. 6.3 (c). Although almost all the corners were found, a linear interpolation was applied to increase the number of points as shown in Fig. 6.3 (d). Once the corners of the projected grid were found, they were related to the picture center and grid center to apply the triangulation. Fig. 6.4 shows the final reconstructed image. The right image shows the reconstructed object, the left image represents the curvature found by the process. The third image present represents the image in 3D space.

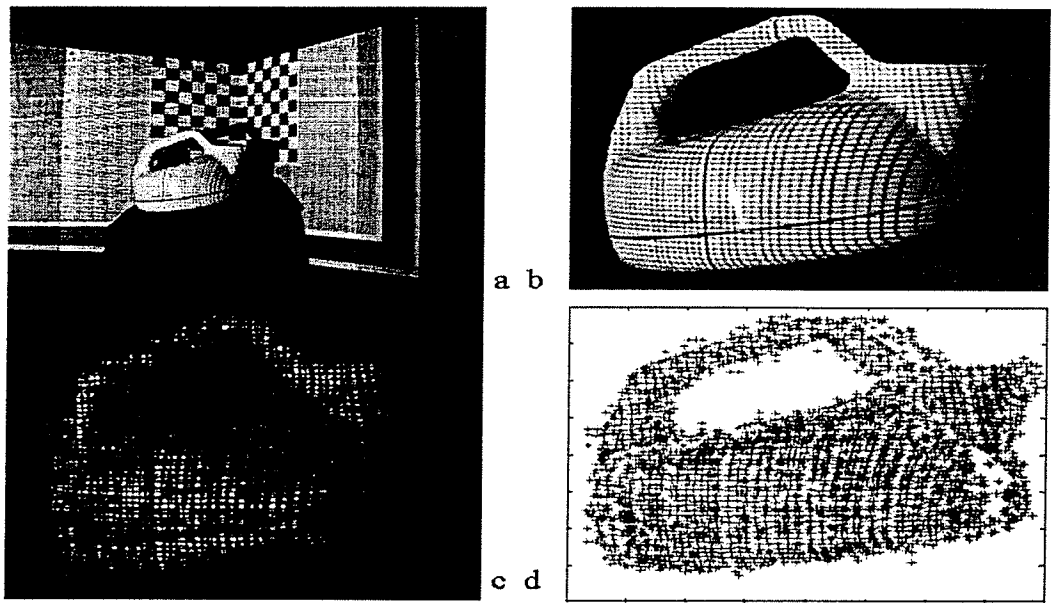


Fig. 6.3 Electric kettle image preparation; a) Image to be reconstructed by the 6X6 projected grid, b) Zoom of the projected mesh, c) corner detection d) interpolation.

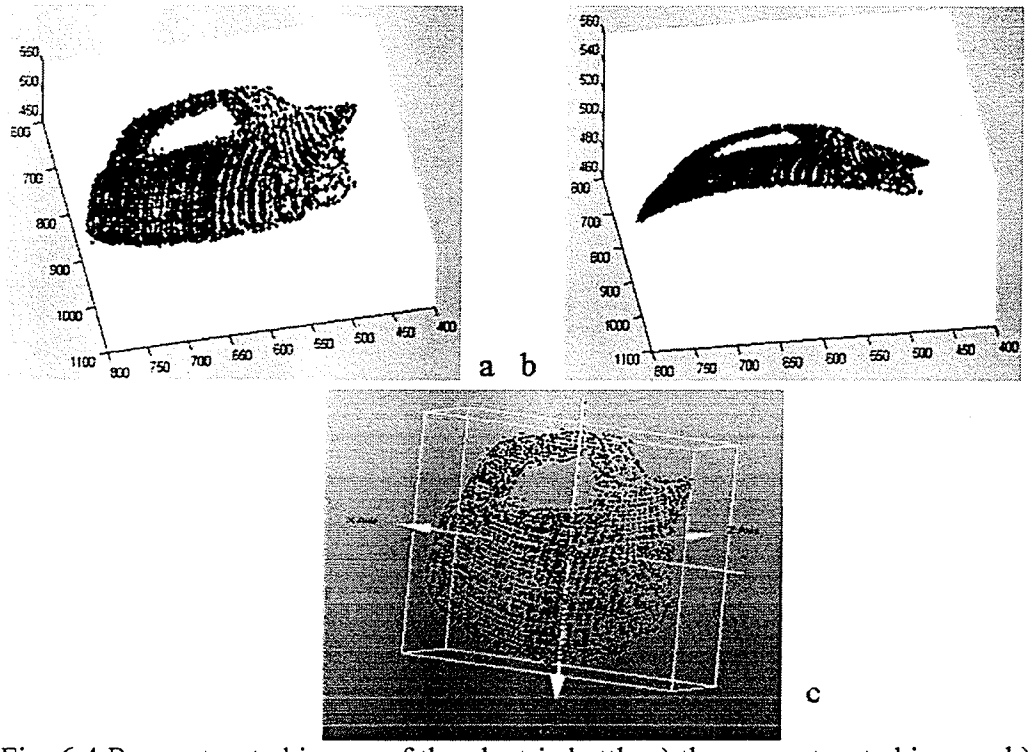


Fig. 6.4 Reconstructed image of the electric kettle a) the reconstructed image, b) curvature of the kettle from the lateral view, c) other view of the result

In order to study the functionality of the developed method, the recovered shape was compared with other sets of results. The first one was the original data obtained by the physical measure of the object. The object was measured manually with a ruler in the different directions. In the case of the depth value, only half of the object was measured since the projected grids are only available to reach this part. The other was a single laser scan as shown in Fig. 6.5.

The single scan was performed to show the same surface acquired with the developed method. The results presented in the Table 6.2, considered the maximum and minimal value direction present in the reconstructed object. The difference between the values was considered to obtain width, height and depth. The error is calculated by the difference between the real dimensions of the object minus the estimated dimension divided by the real dimension

The original values of the teapot are 210mm, 165mm, and 85mm. corresponding to X, Y, Z. The Z value was measured from the half of the object since it is the part that the developed method is able to capture. The error reported corresponding to the Z seems to be wrong. The reason is the points obtained by the laser scanner overpass the manual measured object, where there is only half of the object considered.

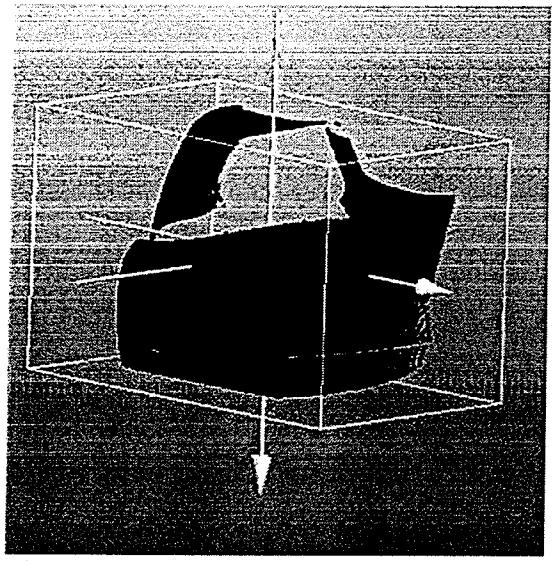


Fig. 6.5 one shot of the laser scanner

Table 6.2 Electric Kettle Reconstructions Comparison (mm)

Developed M.	X	Y	Z	**Laser Scanner**	X	Y	Z
Points reconstructed	2,917	2,917	2,917	Points reconstructed	344,375	344,375	344,375
Min value	101.4	259.9	311.7	Min value	236.50	-73.829	314.28
Max value	299.2	434.9	394.8	Max value	451.37	73.077	439.68
Estimated Dim	197	174.1	83.1	Estimated Dim	214.87	146.90	125.46
Difference with Real object	12.4	9.1	1.9	Difference with Real object	4.87	18.1	40.46
Error Real object (%)	5.9	5.5	2.2	Error Real object (%)	2.31	10.96	47.6

Comparing images of the object captured by the developed method and the laser scanner (Fig. 6.4, 6.5), it is obvious that the laser approach is superior. The scanner-reconstructed image has 118 times more points than the developed algorithm. Therefore the quality is better. Although more points are needed to obtain a better approximation, the developed algorithm cannot afford that. More points to be calculated represents a higher processing time that eventually crashes down the computer due to the limitation of Matlab.

The structured light approach generated a close approximation to the real object. The obtained values of the length, depth and height are a few millimeters over the original values. In Fig. 6.4 c it can be observed that the curvature along the body is similar to the original object. It is also identified the difference between the body and the handle, the same as the body and the spout. The slot between the walls of the spout and the handle are also presented in the reconstructed object.

Dimensions reported by the laser scanner seem to be erroneous in the height and depth. This is due to the acquired points. In the case of the height, the teapot handle was not scanned completely. Therefore fewer points were acquired. For the depth, the scanner was able to obtain more points than the established method for the Z reconstruction where only the half of the object was supposed to be acquired. Therefore more points were presented.

A detail comparison was made between the scanner and the developed method. The data corresponding to each method was compared by overlapping them. The length and height recovered are almost the same but different orientation. The depth comparison presents a big difference. The data captured by the scanner is not the same area captured by the developed method. Even though the orientation when acquiring data tried to be the same in both methods, some differences are presented. At the same time, the laser beam projected by the scanner covers more region of the object than the mesh projected. The curvature presented in the developed algorithm is not as good as the scanned one. The reason is the laser scanner was able to obtain more points from the back part of the

object's body than the established method for reconstruction. Different views of the comparison are presented in Fig. 6.6. The object in black color is the result generated by the scanner. The blue one is the approximation acquired by the developed method.

The second object tested was a volleyball ball. The same steps were followed as the teapot. The diameter of the ball is 200mm. Only one of the faces was considered. Therefore only half of the ball was supposed to be reconstructed. Fig. 6.7 shows the results corresponding to the image acquisition of the projected mesh, the corner detection and the interpolation. Fig. 6.9 shows the obtained approximation.

In Fig. 6.10 the collected data by the laser scanner is shown, making emphasis in the curvature acquired. The comparison with the real object and the laser scanner are presented in Table 6.3

The results generated by comparing the real data were not as good as expected. The shape of the ball was not reconstructed in a correct way. The length and height of the reconstructed part are larger than the original and laser scanner. One reason of the difference can be related to the points calculated. Although the corner detector finds the corners, corners not corresponding to the object are considered. Sometimes the corner detector founds points with different pixel intensity in the scene and marks it like a corner, producing noise in the image.

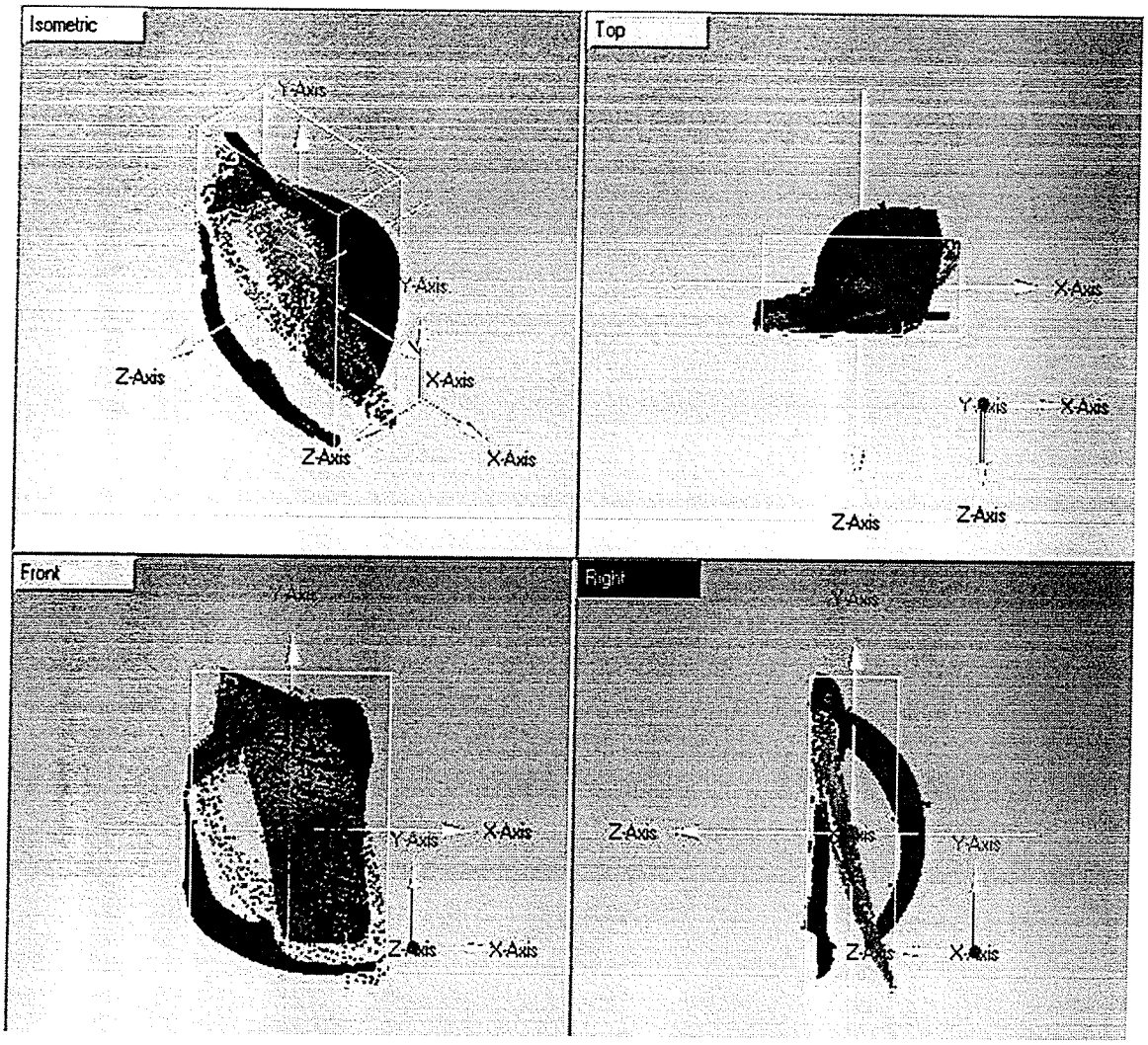


Fig. 6.6 Comparing the object reconstructed with the structured based method and the laser scanner

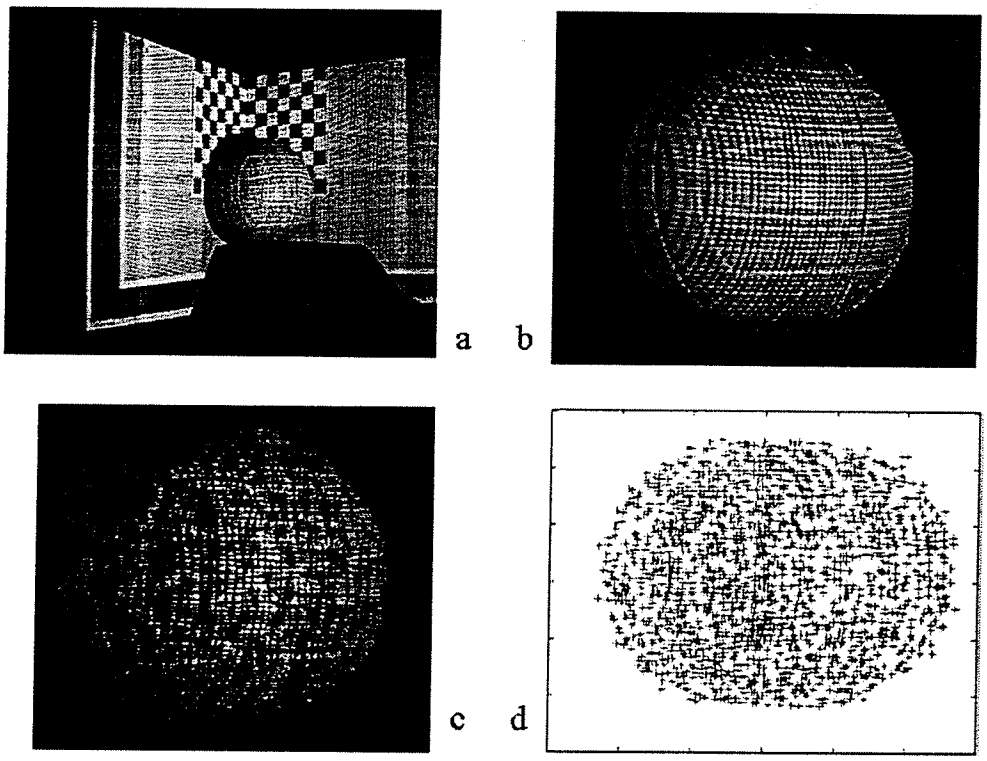


Fig. 6.7 Volleyball ball image preparation a) Image to be reconstructed by the 6X6 projected grid, b) Zoom of the projected mesh c) corner detection d) interpolation

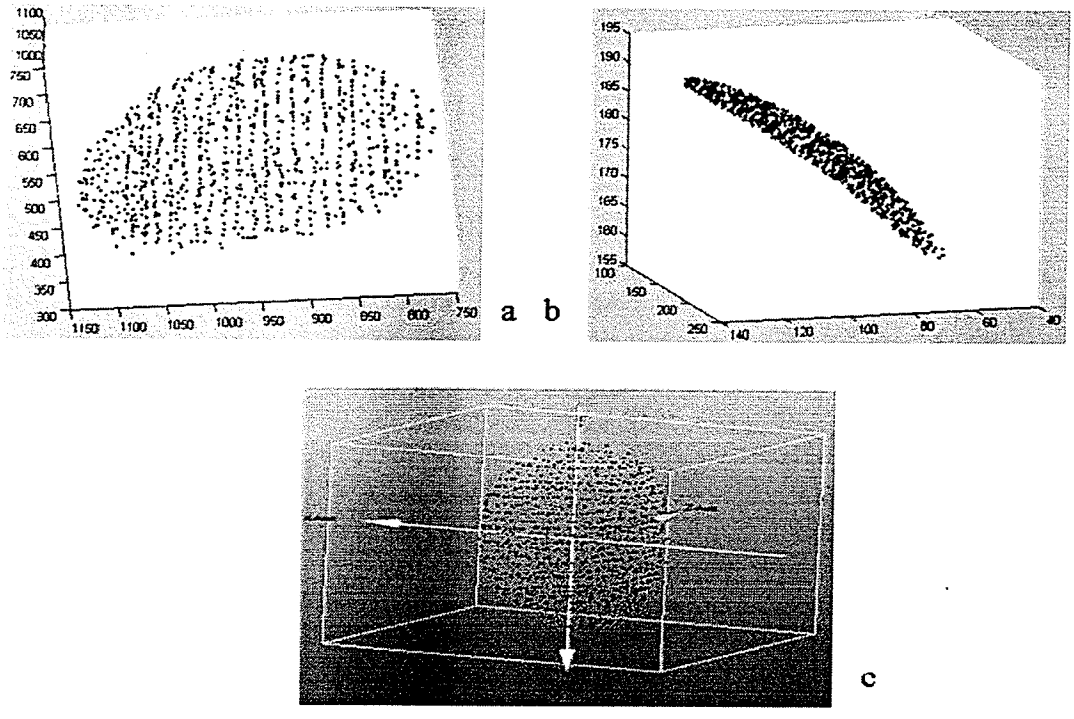


Fig. 6.8 Reconstructed image of the ball a) reconstructed image, b) curvature of the ball, c) other view of the result

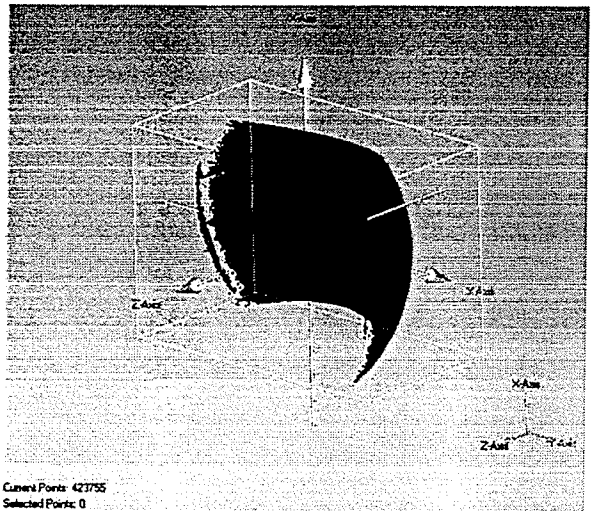


Fig. 6.9 Single shot of a ball using the scanner

Table 6.3 Ball Comparison (mm)

Developed M.	X	Y	Z	**Laser Scanner**	X	Y	Z
Points reconstructed	867	867	867	Points reconstructed	423,755	423,755	423,755
Min value	51.04	121.9	146.8	Min value	-65.14	567.34	282.58
Max value	157.6	215.6	192.7	Max value	71.58	758.67	385.57
Estimated Dim	106.5	93.98	45.9	Total length	143.72	191.33	102.99
Difference with Real object	93.5	106	54.1	Difference with Real object	56.25	8.67	-2.99
Error Real object (%)	46.75	53	27.05	Error Real object (%)	28.12	4.33	NA

The curvature present in the ball is not recovered because the camera did not catch the lines projected. The lines deformed following the curvature of the ball were not present in the image therefore no corners were detected.

In the results corresponding to the laser scanner, some differences are present too. These differences can be produced due to an incomplete reconstruction of the object. Scanning

the object more times will generate a better approximation. The depth reconstructed by the scanner over passed the required scanning area, presents a bigger curvature. The data comparison between the implemented method and the laser scanner was not possible. The difference between the objects is huge, the pictures cannot be overlapped. The value NA present in the error is due to the fact that more points recovered than the points measured in the comparison.

The last object scanned was the shape of a foot. A foot plaster was followed by the same procedure as the last two samples. Fig. 6.10 shows the object acquisition, corner detection and linear interpolation. Fig. 6.11 shows the reconstructed foot by the developed method. Fig. 6.12 shows the data obtained by the laser scanner. Table 6.3 compares the results of the two methods and the original dimensions measured directly from the object. These are 280 mm length, 87 mm width and 90 mm depth.

In the case of the foot, the results seem to fit better than the other two objects. The difference in length, height and depth is minimal compared with the original object. The toe of the foot is getting a curvature because the position of the image scanned. The picture taken is not able to see the complete lines projected. Better orientation of the object to acquire could generate better results.

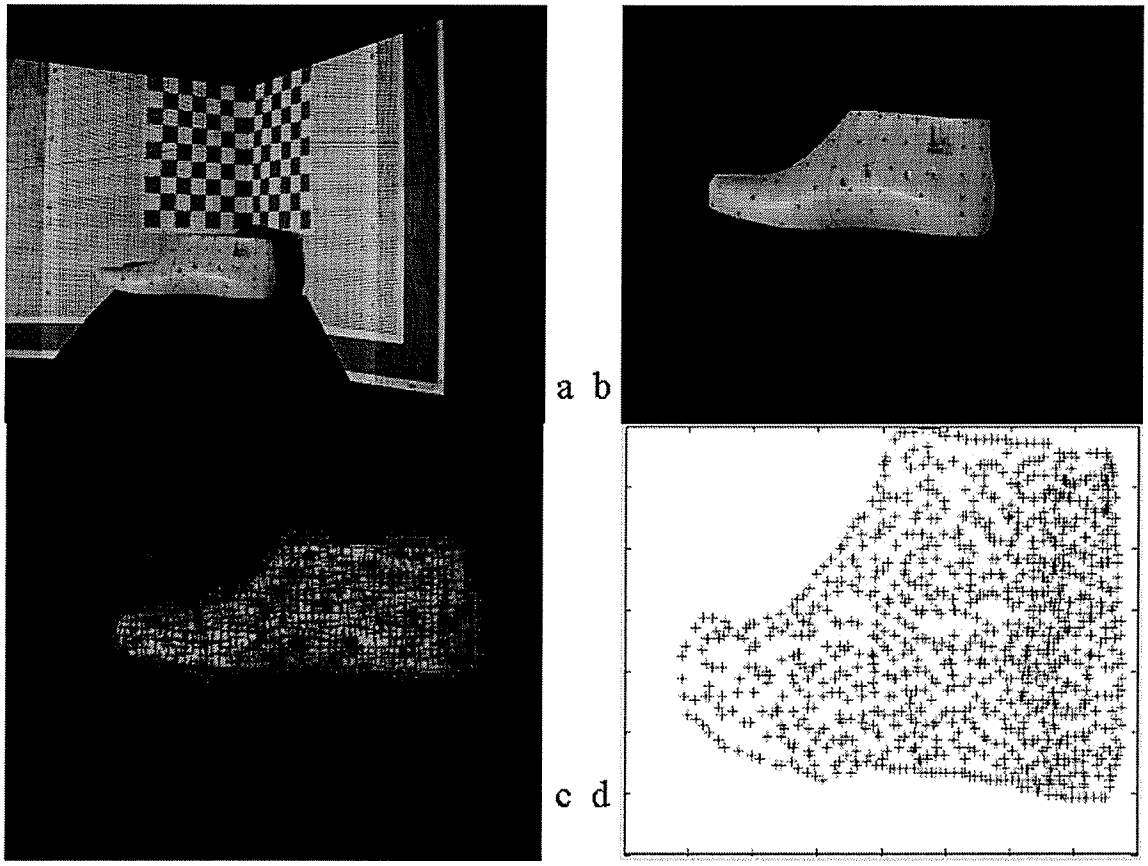


Fig. 6.10 Foot image preparation a) Image to be reconstructed by the 6X6 projected grid, b) Zoom of the projected mesh, c) corner detection d) interpolation

The shape acquired by the scanner looks similar to the acquired shape by the developed method. An obvious difference is the number of points reconstructed. The laser scanner has 490 times more points that the proposed method, this is the reason why it looks darker.

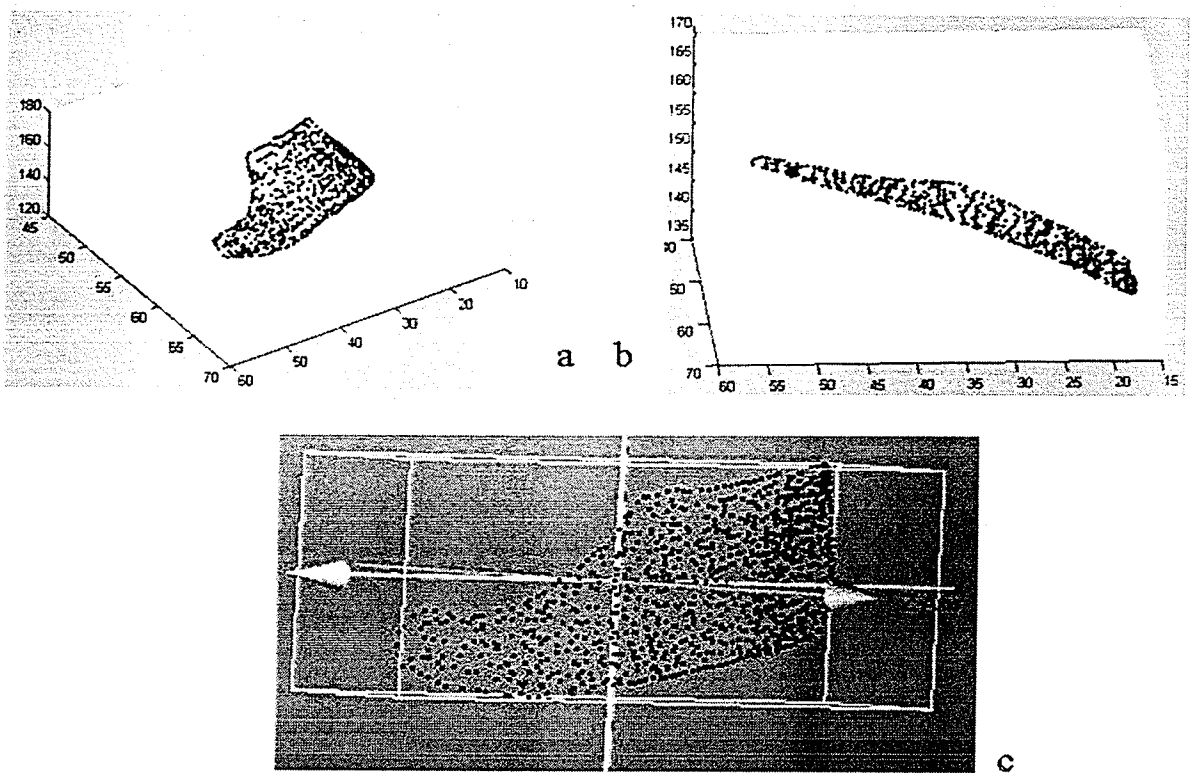


Fig. 6.11 Reconstructed image of the foot, a) reconstructed image, b) curvature of the teapot, c) other view of the result

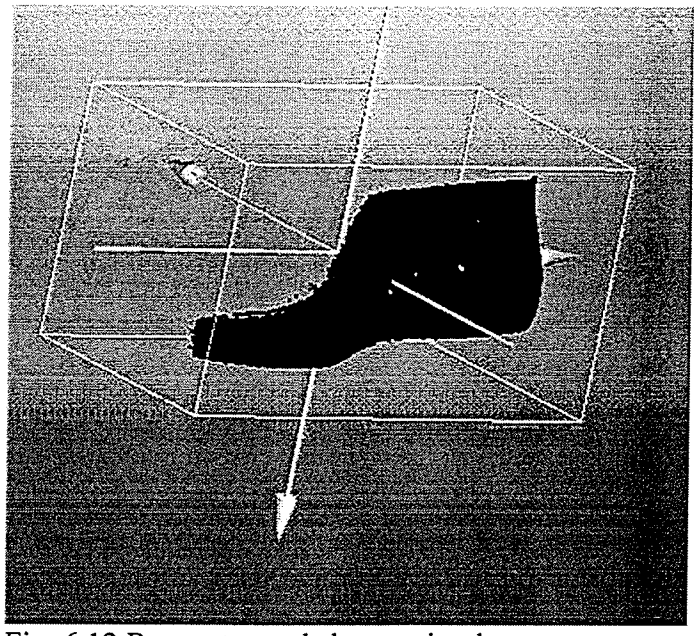


Fig. 6.12 Reconstructed shape using laser scanner

The comparison between the results of the two methods was also performed and results are presented in Table 6.4. The difference between the laser scanner and the real dimensions is minimal. One of the reasons why the foot seems to be smaller is the lack of data from the acquired corners. The threshold the image processing deleted some of the grid lines closed to the borders because of different image background, which results in missed corners. Once again the depth reconstructed by the scanner over passed the depth measured from the object for the reconstruction. Even the resulting data is smaller than the scanned one; the data seems to overlap in a correct way as shown in Fig. 6.14.

Table 6.4 Foot Comparison (mm)

Developed M.	X	Y	Z	**Laser Scanner**	X	Y	Z
Points reconstructed	671	671	671	Points reconstructed	328635	328635	328635
Min value	56.9	142.7	326.7	Min value	466.20	-45.86	283.68
Max value	295.9	234.4	388.1	Max value	737.38	50.35	351.24
Estimated Dim	239	91.7	61.4	Total length	271	96.21	67.56
Difference with Real object	41	4.7	28.6	Difference with Real object	9	9.21	22.4
Error Real object (%)	14.6	5.4	31.7	Error Real object (%)	3.21	10.6	24.3

The shape of the foot was reconstructed as one of the objectives of the research. One of the characteristics of the foot is that it has only some curvatures in the case of the arc and in the ends.

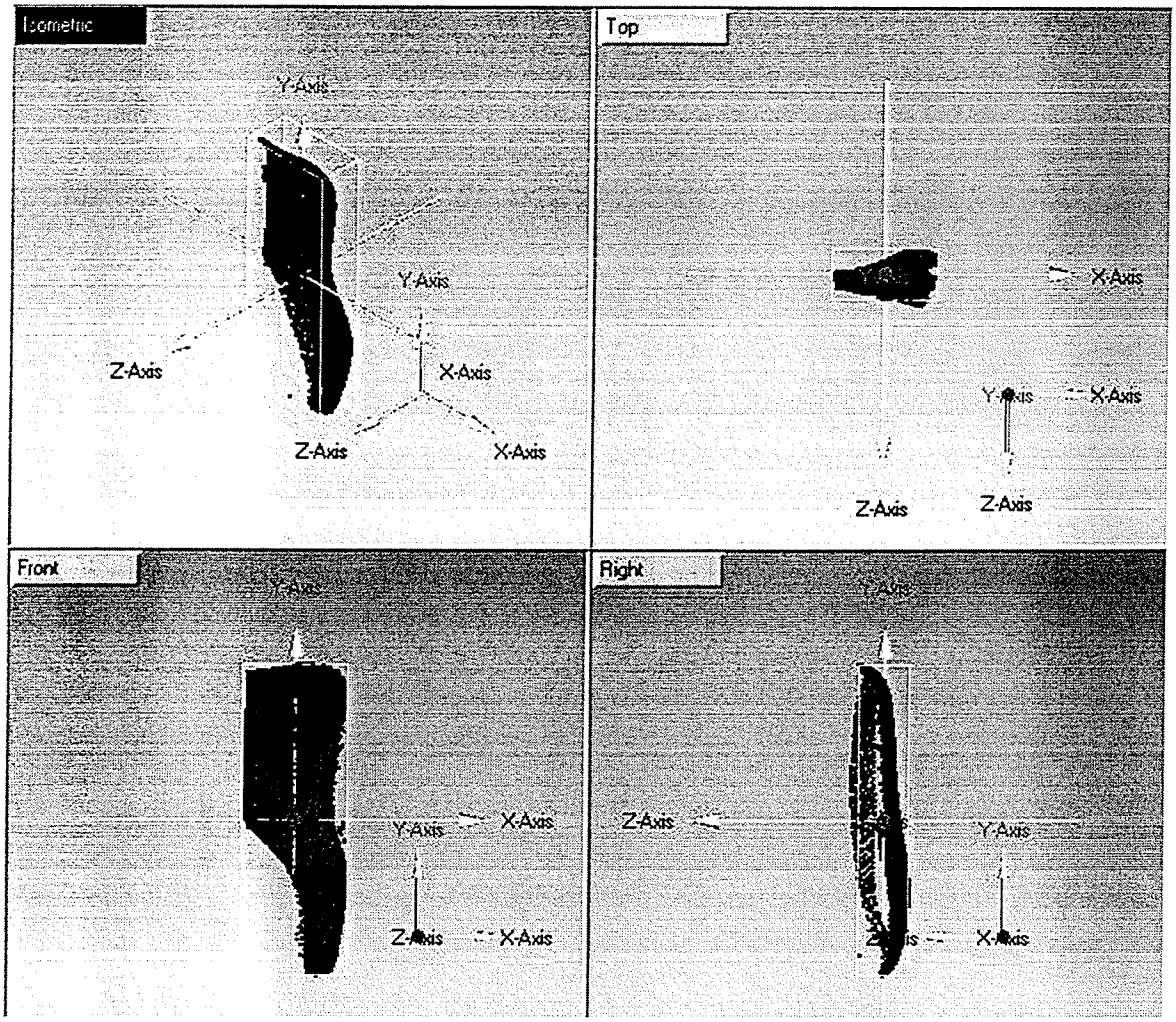


Fig. 6.13 Comparing the object reconstructed by the developed method and the laser scanner

These examples demonstrated the performance of the developed method. The results presented were compared with the real data measured directly from the object and the data generated from a laser scanner. The introduced method generates a good approximation in the length and height. Some difficulties were presented when reconstructing depth, especially if the object is curved. The major problem in curvatures was shown in the ball. In this example the projected lines are lost or become smaller in curve areas, making the reconstruction a complex process. Smooth curvatures are well

reconstructed, as the case of the teapot. Linear objects are well recognized and reconstructed as the case of the foot. Lines are well identified making the process effective. Problems related to accuracy in the length and high could be fixed by focusing more on borders of the acquired image. The values obtained by the developed method seem better than the values generated by the scanner. The reason is that the surface obtained by the scanner is not the same one acquired by the developed method. The orientation varies at the moment of getting the data. Other factor is the real measurement of the object. The measurement of the object was from the orientation of the captured image by the developed method.

In the case of concave objects or surfaces with holes, the algorithm will not work. The reason is that the projected lines are not present in the holes, making the reconstruction process impossible to achieve.

Chapter 7. Conclusions and further work

7.1 Conclusions

This research was focused on finding a cost-effective and fast 3D reconstruction approach. Different methods were introduced to study the existing methods used for 3D acquisition. The methods such as the laser scanner, CMM, SFS, stereovision and structured light were reviewed and compared. Considering the applications established for the project in the introduction, two methods where selected from these revisions.

The characteristic of the selected methods is to capture the object under reconstruction using a single camera. The methods are SFS and the structured light approaches. SFS is the method where the depth reconstruction depends on an adequate light orientation. Structured light approach is where a light with known characteristics such as measure and colour is projected onto the object to obtain the 3D data.

SFS method was first considered due to its simple and fast reconstruction. Different techniques related to this approach were studied. Two of these techniques were adopted due to the good results reported by the authors and the easy implementation of the algorithms. The two methods were tested and compared using synthetic and real images. According to the results produced, a method suitable for modifications was selected. The first method analyzed was Pentland's algorithm, and another was Tsai and Shah's method.

Pentland's method produces good results on most surfaces that change linearly. It was found that this approach produces good results for real images. Tsai and Shah's approach was selected due to the simplicity of the algorithm and the good results it generated. The algorithm presents a better approximation and the process is faster than other methods tested. The reconstructed images seem to be smoother than Pentland's approach. Therefore, this method was selected for improvements.

The improvements were in the image processing before using the algorithm. Image enhancement and filtering were applied obtaining better results than the studied algorithms. The modified algorithm presents a better-reconstructed image since the surface seems to be smoother, and approximates to the original value.

Although the shape reconstructed by the modified method was better compared to the original Tsai and Shah's and Pentland's methods, the dimensions obtained were still not accurate enough for the application. Another problem was related to the illumination conditions required for the reconstruction. It is required to establish a good distribution of the lights surrounding the desired object to be scanned. Different position and intensity of the lights were required to be adjusted to obtain the perfect illumination. Brightness parts are not recovered at all. For these reasons and the fact that the assisting light has to be used to obtain a better result. This method was not suitable for the objectives of the research. Therefore it was concluded that a new method has to be considered where a static light is required to avoid the problems presented in SFS approach.

The developed approach follows the restrictions established for the research where only a camera and a source of light are allowed. This method is based on structured light and triangulation. A complete non-coloured grid is projected onto whole surface of the object in the proposed method to simplify the image capturing and processing. The traditional structured light methods use strips of different grey intensities or bands with colours.

The basic idea of the proposed approach is to take a picture of the deformed grids projected onto the object by the projector. The projected lines are shifted or deformed on the object surface, which is captured on the image. This relation is used to find the 3D position of each recovered point of the object using triangulation. A calibration of the camera and projector used is required to relate the projected pattern with the captured image.

Three objects were tested to evaluate the proposed approach. The first object tested was a teapot, the second is a ball and finally a foot plaster. The recovered data were compared with the same object data but measured by different methods. One group of data was obtained by a manual measurement using the standard ruler. Another group of data was obtained by the laser scanner.

The developed method generates a good approximation in the length and height. Some difficulties were presented when reconstructing depth especially if the object is curved. The problem is related to the projected lines. Lines are lost or become smaller in curved areas, making the reconstruction a complex process. In addition to this, the camera is not

able to capture some of the features behind a curve. The angle of vision of the camera cannot reach some parts of the curve, the same as the projected grids. On the other hand, smooth curvatures are well reconstructed since there is little change of the projected lines on the object. Linear objects are well recognized and reconstructed in the case of the foot model. Lines are well identified making the process effective. Although the data reconstructed is not as accurate as the one acquired by a laser scanner, it is a good approximation to real object in the case of the foot model. There is no doubt that this method generates a better approximation than SFS approach.

The objectives of the project have been achieved in the research. An evaluation of 3D acquisition methods was performed. A cost-effective method for 3D data acquisition has been developed. A comparison was performed to evaluate the performance of the proposed method. The proposed approach is easy to use and the cost-effective comparing with 3D laser scanners.

7.2 Further work

To improve accuracy of the proposed method, one of the areas that must be improved is to increase the number of points processed. In the proposed method the number of points considered is not enough to make the surface smooth. Although points increase in the point density was attempted, the existing program does not support the data. The program could be re-written in other programming language such as C or C++ to fix this problem.

Problems related to precision could also be fixed by focusing more on the borders of the acquired image. Instead of corner detection, edge detection could be performed first to find the complete lines across the object.

Curvatures also present a problem that must be fixed. A different position of the object or the projector could be a solution of this problem. Although not all the curvatures could be reconstructed, applying these changes could generate a better result.

A high resolution of the picture can also improve the result of the developed method. In a better image, a better corner detection can be found. In the same way if the projector is substituted or located in other orientation a better approximation could be achieved.

Another way to improve the accuracy is by locating more precise lines projected onto the object. An idea is to project the grid with different colours in each line. In this way, each color can be tracked along the surface to be reconstructed.

In this research the comparison between the developed method and the laser scanner was performed. The values compared were obtained by the difference of the minimal and maximal values present in the axis of interest. The results reported are not highly accurate because only the minimal and maximal values are considered in comparison. An algorithm which can compare all data obtained has to be implemented. Therefore, other way to measure and compare the data obtained by both methods has to be implemented.

One path to follow is to find a common point along the data of the two methods. The coordinates of the points have to be known and related to established common position. At this moment only the x and y values are considered to locate the point. Once the point correspondent to each data is correctly found in the same position, the depth value is analyzed. A comparison between the points could be performed to obtain the real

difference between them. For each point an error can be established by a normalization of the depth data corresponding to the scanner and then compared with the depth of the developed method. This process could be repeated for all the points found in the developed method. The reason is that the developed method has fewer points than the data generated by the laser scanner

A combination of SFS and the developed method could be considered for further work as well. The only restriction to be considered is the fact that the lights required by the SFS method should be in some way absorbed by the surface to have not noise induced by them. The object should be covered or coloured to reduce the effect of reflection caused by the lights.

Another area that may be interesting to follow is the creation of a prototype for the foot clinic. A compact system must group the projector and the camera. An idea of this prototype is presented next.

The projector required must be small and cheap. The projector will only be used for projecting the grid. The proposed projector is a ***Sony VPLES-1.*** Th e price of this projector is 1499.99 Canadian dollars. It was selected based on the price and the size. The dimensions of this projector are 22.8 cm width, 6.6 cm high and 28.5 cm depth. Fig. 7.1 shows the projector.

It is also recommended that the camera import the pictures automatically to the computer for processing. Therefore a web-camera is proposed. The camera must have high resolution as the camera used for this research. The camera recommended is the "***Logitech Quickcam Pro 4000 Pc Camera***" that has a value of 159.95 Canadian dollars. The resolution of the camera is 1280X960 pixels, 1.3M pixels. Fig. 7.2 shows the camera proposed.

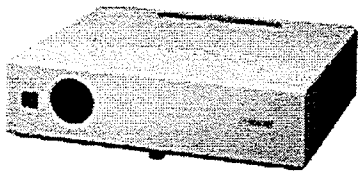


Fig. 7.1 Proposed Projector

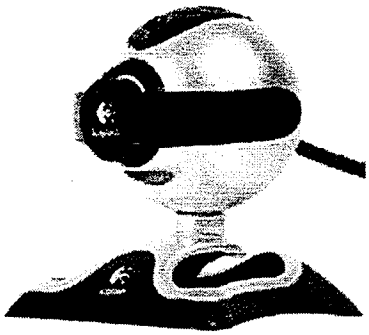


Fig. 7.2 Proposed Camera

The design of the system is proposed as shown in Figs. 7.3 and 7.4. The projector is placed into a case. The camera is located in the left of the projector. The camera lens is located at the same height of the projector. An angle is present in the camera but facing the right size of the projector. A platform is also required to support the foot. A black structure recommended preventing reflection on the walls. A sketch of the proposed head of the system is shown in Fig. 7.3. A complete overview of the structure is presented in Fig. 7.4.

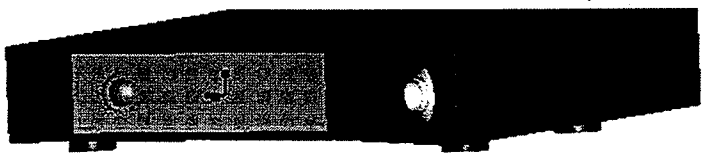


Fig. 7.3 Proposed head of the System

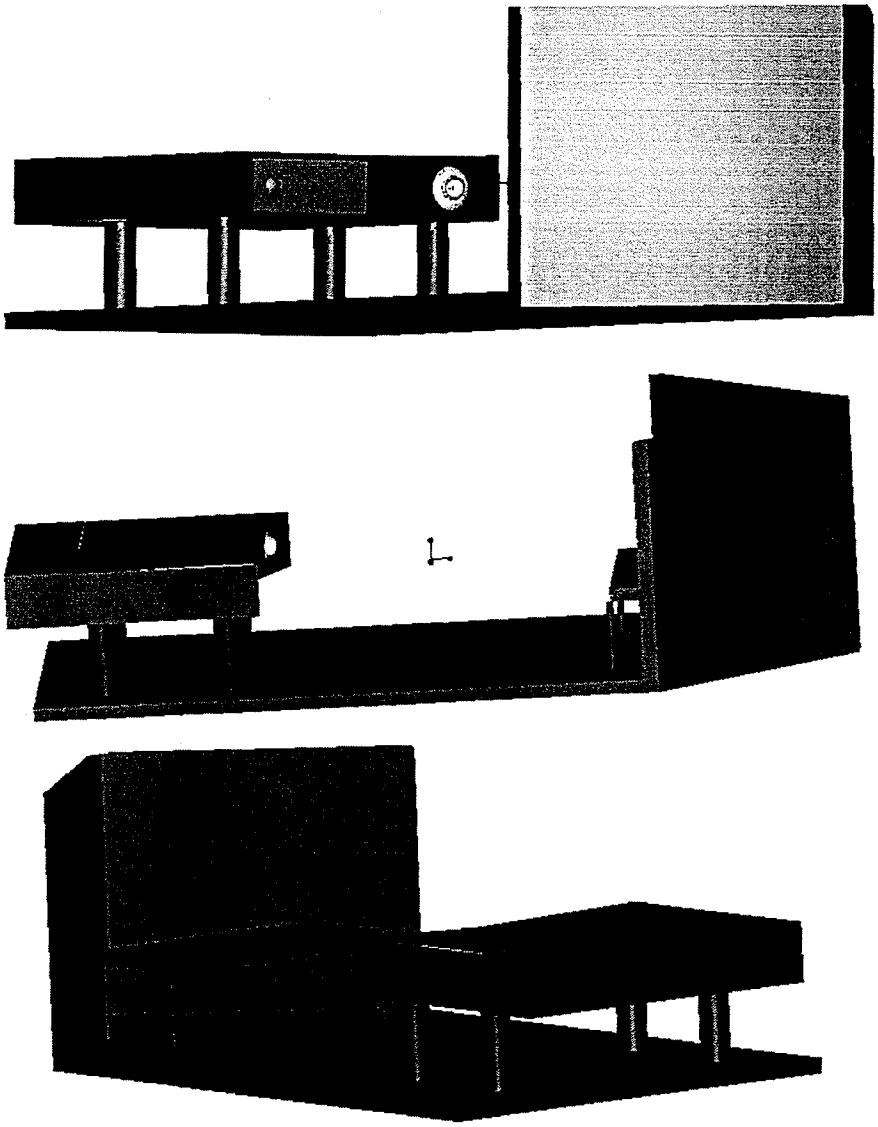


Fig. 7.4 Complete overview of the proposed structure

References

[1] Ayache, N. and Lustman, F. "Trinocular stereo vision for robotics". IEEE *Transactions on pattern analysis and machine intelligence, Vol. 13, No.1* (1991): 73-84

[2] Beyer, H.A. "Accurate calibration of CCD-Cameras". *International conference on Computer Vision and Pattern Recognition, IEEE* (1992): 96-101

[3] Mahadevan, S., Pandzo, H., Bennamoun, M. and Williams, J.A. "A 3D acquisition and modeling system" *Space Centre for Satellite Navigation, Queensland University of technology, IEEE*, (2001): 1941-1944

[4] Kiyasu, S Hoshino., H Yano., K. and Fujimura, S. "Measurement of the 3-D shape of specular polyhedrons using an m-array coded light source*" IEEE Transactions on Instrumentation and Measurement* (1995): 775–778.

[5] ISO/DIS 14253-2 "Guide to the estimation of uncertainty in GPS measurement, in calibration of measuring equipment and in product verification". (1998) < http://www.iso.org>

[6] Brenner, C. Guhring,J. Bohm, J. and Fritshch, D. "Data processing and calibration of a cross pattern stripe projector". *IAPRS, Vol. XXXIII, Amsterdam*, (2000)

[7] Horn B.K.P., "Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View", *Massachusetts Inst of Technology, Artificial Intelligence Laboratory*, (1970)

[8] Bergmann, D. "New approach for automatic surface reconstruction with coded light" *Proceedings of Remote Sensing and Reconstruction for Three-Dimensional Objects and Scenes, Vol. 2572, SPIE,* (1995): 2–9.

[9] Caspi, D., Kiryati, N. and Shamir J. "Range imaging with adaptive color structured light" *Pattern analysis and machine intelligence* (1998): 470–480.

[10] Chen, C. Hung, Y. Chiang, C. and Wu J. "Range data acquisition using color structured lighting and stereo vision" *Image and Vision Computing* 15 (1997): 445–456.

[11] Gühring, J. "Dense 3-D surface acquisition by structured light using off-the-shelf Components" *Video metrics and Optical Methods for 3D Shape Measurement 4309* (2001): 220–231.

[12] Kiyasu, S Hoshino., H Yano., K. and Fujimura., S. "Measurement of the 3-D shape of specular polyhedrons using an m-array coded light source" *IEEE Transactions*

on Instrumentation and Measurement (1995): 775–778.

[13] Horn, B.K.P, "Robot Vision" *The MIT Press, McGraw-Hill* (1987).

[14] Lee C.H. and Rosefeld, A. "Improved Methods of Estimating Shape from Shading Using the Light Source Coordinate System," *Artificial Intelligence, Vol.26, No. 2,* (1985): 125-143.

[15] Pentland, A. "Shape information from Shading: A theory about Human Perception", *Proc. Int'l Conf. Computer Vision*, (1988): 404-413.

[16] Bichsel, M. and Pentland, A.P. "A Simple Algorithm for Shape from Shading." *IEEE Proc Computer Vision and Pattern Recognition*, (1992): 459-465.

[17] Tsai, P.S and Shah, M. "Shape from shading using linear approximation," *Image and Vision Computing Vol.12, no.8,* (1994): 457-498

[18] Kimmel, R. and Bruckstein, A. "Global Shape from Shading" *Computer Vision and Image Understanding, Vol.62 No.3* (1995): 360-369.

[19] Tsai, P-S., Shaha, M., Zhang, R. and Cryer, J.E. "Analysis of Shape from Shading Techniques" *Computer Vision and Pattern Recognition*, (1994): 377-384.

[20] Fanany, M.I. and Kumazawa, I. "Analysis of shape from shading algorithms for fast and realistic 3D face reconstruction" *IEEE Proc. APCCAS* (2002): 181-185

[21] Bakhadyrov, I. and Jafari M. "Snake-based deformable surface model for shape from shading" *IEEE international Conference on Systems, Man and Cybernetics*, (2002)

[22] Pentland, A. "Local Shading Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 6, No. 2,* (1984): 170-187.

[23] Pentland, A., "Linear Shape from Shading", *International Journal of Computer Vision Vol.4* (1990): 153-162.

[24] Petriu, E. M. Bieseman T., Trif N., McMath, W. S. and Yeung K. "Visual object recognition using pseudo-random grid coding", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (1992): 1617–1624.

[25] Posdamer J. L. and Altschuler M. D. "Surface measurement by space-encoded projected beam systems", *Computer Graphics and Image Processing* (1982): 1–17.

[26] Inokuchi S., Sato K., Matsuda F. "Range imaging system for 3-D object

recognition". *Proceedings of the International Conference on Pattern Recognition*, (1984): 806–808.

[27] Trobina, M. "Error model of a coded-light range sensor". *Technical report, Communication Technology Laboratory, ETH Zentrum, Zurich* (1995). <http://citeseer.ist.psu.edu/97770.html>

[28] Chen, C. Hung, Y. Chiang, C. and Wu J. "Range data acquisition using color structured lighting and stereo vision" *Image and Vision Computing* (1997): 445–456.

[29] Sato, K. "Range imaging based on moving pattern light and spatial-temporal matched Filter". *IEEE International Conference on Image Processing, Vol. 1*, (1996): 33–36.

[30] Hall-Holt, O. and Rusinkiewicz, S. "Stripe Boundary codes for real time structures light range scanning of moving objects". *Stanford computer graphics labs, Stanford University* (2003)

[31] Maruyama, M. Abe, S. "Range sensing by projecting multiple slits with random cuts *Pattern Analysis and Machine Intelligence 15* (1993): 647–651.

[32] Durdle, N. G Thayyoor,. J. and Raso V. J. "An improved structured light technique for surface reconstruction of the human trunk" *IEEE Canadian Conference on Electrical and Computer Engineering, Vol. 2,* (1998): 874–877.

[33] Morita, H. Yajima K., and Sakata S. "Reconstruction of surfaces of 3-D objects by m-array pattern projection method" *IEEE International Conference on Computer Vision,* (1988): 468–473.

[34] Morano, R. A. Ozturk C., Conn, R. Dubin, S Zietz. S., and Nissanov, J. Structured light using pseudorandom codes", *Pattern Analysis and Machine Intelligence* (1998): 322–327

[35] Carrihill B., Hummel R. "Experiments with the intensity ratio depth sensor". *Computer Vision, Graphics and Image Processing, Vol. 32, Academic Press* (1985): 337–358.

[36] Miyasaka, T. Kuroda K., Hirose, M. and Araki, K. "High speed 3-D measurement system using incoherent light source for human performance analysis" *Proceedings of the 19th Congress of The International Society for Photogrammetry and Remote Sensing,* (2000): 65–69

[37] Tajima J. and Iwakawa, M. "3-D data acquisition by rainbow range finder". *International Conference on Pattern Recognition*, (1990): 309–313

[38] Geng, Z. J. "Rainbow 3-dimensional camera: New concept of high-speed 3-dimensional vision systems" *Optical Engineering* (1996): 376–383.

[39] Gonazalez, R.C.,Woods, R.E. "Digital Image Processing". *Second edition, Prentice Hall* (2002).

[40] Harris, C.G. and Stephens M.J. "A combined corner and edge detector". *Proceedings Fourth Alvey Vision Conference, Manchester.* (1988): 147-151

[41] Raskar, R. and Beardsley, P. "A self-correcting projector", *Mitsubishi Electric Research Laboratories (MERL), IEEE* (2001): 504-508

[42] Faugeras, O. "Three-Dimensional Computer Vision: A Geometric Viewpoint". *Cambridge , Mass. :The MIT Press* (1993)

[43] Tsai R.Y. "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. RA-3. No.4* (1987)

[44] Marchessoux, C., Richard, N and Fernandez, C. "Description of a simple method in 3D reconstruction in medical imaging" *Proceedings of the First International Symposium on 3D Data Processing Visualization and Transmission IEEE* (2002)

[44] Hanson, A.R., "Introduction to Computer Vision, Image Formation", *University of Massachusetts Amherst,* (2002).

[45] Haralick, R.M. and Shapiro, L.G. "Computer and robot Vision", *Vol. II Addison-Wesley Publishing Company* (1993)

[46] Weng, J., Cohen, P. and Herniou, M. "Camera Calibration with distortion models and accuracy evaluation". *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, no 10,* (1982): 965-980

[47] Valkenburg, R.J., and McIvor, A.M. "Acurate 3D measurement using structured Light System" *Imaging and Sensing Industrial Research Ltd.* (1997) < http://citeseer.ist.psu.edu/225156.html>

[48] Zhang. Z. "A Flexible New Technique for Camera Calibration" *Technical Report MSR-TR-98-71, Microsoft Corporation* (1998) <http://www.research.microsoft.com/%7Ezhang/Papers/TR98-71.pdf>

[49] Bouguet, J-Y Camera Calibration Toolbox for Matlab (2004), <http://www.vision.caltech.edu/bouguetj/calib_doc/>

Appendix A. Pentland Matlab code

```
%Pentland.m

Ima=imread('image.jpg');                    %Read the image
imshow(Ima)                                 %Display the image


%This image is a 8 array and a double is needed, for that:
%convert the unit 8 array to double array and remove the Z

Imac=double(Ima(:,:,1));;
[nrow,ncol]=size(Imac);
Im=Imac/255;


lg=[.866,1.5,1];                            %Ligth direction Sx, Sy, Sz;
lg2=lg/norm(lg);
tilt = atan2(lg2(2),lg2(1));
slant = acos(lg2(3));
costilt=cos(tilt);
sintilt=sin(tilt);
sinslant=sin(slant);

FI=fft2(Imac/255);                          %Applying Fourier
for r=1:nrow
   for c=1:ncol
      if r==1 & c==1
         FZ(r,c)=0;
      else
        fZ(r,c)= FI(r,c)/ (((c-1)*costilt*sinslant + (r-1))*sintilt*sinslant);

      end
    end
end
Z=real(ifft2(FZ));
```

Appendix B. Tsai and Shah Matlab code

Tsai.m

```
Ima=imread('image.jpg');                    %Read the image
imshow(Ima);                                %Display the image

%This image is a 8 array and a double is needed, for that:
%convert the unit 8 array to double array and remove the Z

Imac=double(Ima(:,:,1));
[nrow,ncol]=size(Imac);
lg=[.866,1.5,1];                            %Ligth direction Sx, Sy, Sz;
lg2=lg/norm(lg);
Iter=2;
W=1e-8;

%Getting points
z_old=zeros(nrow,ncol);
s_old=ones(nrow,ncol);
p=zeros(nrow,ncol);
q=zeros(nrow,ncol);
ps=lg2(2)/lg2(3);
qs=lg2(1)/lg2(3);
ss=(1+ps*ps+qs*qs);

run=1;                                      % Star cycle
iterations=10;
while run

% compute the differences
   p=zeros(nrow,ncol);
   q=zeros(nrow,ncol);
   p(2:nrow,2:ncol)=z_old(2:nrow,2:ncol)-z_old(2:nrow,1:ncol-1);
   q(2:nrow,2:ncol)=z_old(2:nrow,2:ncol)-z_old(1:nrow-1,2:ncol);
   pq  = p.*p + q.*q + 1;
   pqs = p*ps + q*qs + 1;
   f   = (Imac/255 - max(0,pqs./(sqrt(pq).*sqrt(ss))));
   dfz = -( (ps+qs)./(sqrt(ss).*sqrt(pq)) - (p+q).* pqs ./ (sqrt(pq.^3).* sqrt(ss)) );
   K   = s_old .* dfz./( W + (dfz.^2) .* s_old);
   s=(1-K.*dfz).*s_old;
   z=z_old - K .* f;
   z_old=z;
   s_old=s;
   iterations=iterations+1;
    maxs=max(max(s));
    maxf=max(max(f));
   if iterations >=Iter
      run=0;
   end
end
```

Appendix C. Modified Matlab code

ImprovedSFS.m

```
Ima=imread('image.jpg');                    %Read the image
imshow(Ima)                                 %Show the image

%This image is a 8 array and a double is needed, for that:
%convert the unit 8 array to double array and remove the Z

Imac1=double(Ima(:,:,1));
Imac2=ind2gray(Imac1);                      %Converting to gray scale
Imac3=imadjust(Imac2,[],[],.5);             %Adjusting the image
Imac=medfilt(Ima3.[3,3])                    %Applying Median Filter
[a,b]=size(Imac1);
lg=[.866,1.5,1];                            %Ligth direction Sx, Sy, Sz;
lg2=lg/norm(lg);
Iter=10;
end
W=1e-8;

%Getting points
z_old=zeros(a,b);
s_old=ones(a,b);
p=zeros(a,b);
q=zeros(a,b);
ps=lg2(2)/lg2(3);
qs=lg2(1)/lg2(3);
ss=(1+ps*ps+qs*qs);
run=1;                                      %star cycle
iterations=0;
while run

% compute the differences
p=zeros(a,b);
   q=zeros(a,b);
   p(2:a,2:b)=z_old(2:a,2:b)-z_old(2:a,1:b-1);
   q(2:a,2:b)=z_old(2:a,2:b)-z_old(1:a-1,2:b);
   pq  = p.*p + q.*q + 1;
   pqs = p*ps + q*qs + 1;
   f   = (Imac1/255 - max(0,pqs./(sqrt(pq).*sqrt(ss))));
   dfz = -( (ps+qs)./(sqrt(ss).*sqrt(pq)) - (p+q).* pqs ./ (sqrt(pq.^3).* sqrt(ss)) );
   K   = s_old .* dfz./( W + (dfz.^2) .* s_old);
   s=(1-K.*dfz).*s_old;
   z=z_old - K .* f;
   z_old=z;
   s_old=s;
   iterations=iterations+1;
   maxs=max(max(s));
   maxf=max(max(f));
   if iterations >=Iter
      run=0;
   end
end
```

Appendix D. Corner Detection Matlab code

Corner.m

```
%************************CORNER DETECTION*******************************
%This algorithm finds the corners of the images using the using Harris corner Detector. The inputs are the
%pictures of the calibration patterns (camera and projector calibration patterns) as well as the objected to
%be reconstructed. The output is the location of the corners.

function [cornpos,cornerPpict]=corner(Ima)

figure,
imshow(Ima),
title('Real Image to Reconstruct')                  %Show Original Image
RGB = imadjust(Ima,[.2 .3 0; .6 .7 1],[]);          %Adjust Color in the image
I=rgb2gray(RGB);                                    %Transform color Image to grayscale

sigma=4;
thresh=1500
radius=7;
disp=1;


   dx = [-1 1 1; -1 1 1; -1 1 1];                   % Derivative masks
   dy = dx';
   Ix = conv2(im, dx, 'same');                      % Image derivatives
   Iy = conv2(im, dy, 'same');

%Gaussian filter of size 6*sigma (+/- 3sigma) and of minimum size 1x1.

g = fspecial('gaussian',max(1,fix(6*sigma)), sigma);
   Ix2 = conv2(Ix.^2, g, 'same'); % Smoothed squared image derivatives
   Iy2 = conv2(Iy.^2, g, 'same');
   Ixy = conv2(Ix.*Iy, g, 'same');

%Compute the Harris corner measure

k = 0.04;
cim = (Ix2.*Iy2 - Ixy.^2) - k*(Ix2 + Iy2).^2; % Harris

%Extract local maxima by performing a grey scale morphological ,
%dilation and then finding points in the corner strength image that
%match the dilated image and are also greater than the threshold.

   sze = 2*radius+1;                                % Size of mask.
mx = ordfilt2(cim,sze^2,ones(sze));                 % Grey-scale dilation.
cim = (cim==mx)&(cim>thresh);                       % Find maxima.
[r,c] = find(cim);                                  % Find row (y),col(x) coords.
cornpos=[c,r];                                      % Values of the corners detected...x , y
mval=mean(cornval);                                 % Mean value of the corners detected (to
                                                    % find the center of the line)
medval=median(cornval);                             % Median value of the corners detected (to
                                                    % find the centrer of the line)
```

```
% Plot all corners
figure,imagesc(im), colormap gray, title('6. corners'), axis equal, hold on;
h=plot(c,r,'r+');set(h,'linewidth',2.5);
hold off;
WCent=size(Ima)/2; %center of the image.
figure,imagesc(im), colormap gray, title('corners and center'), axis equal, hold on;
h=plot(c,r,'r+');
set(h,'linewidth',2.5)
plot(WCent(:,2),WCent(:,1),'yd')
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\xycamera.mat'
                                        save(savefile, 'corners');
```

Appendix E. Camera and Projector Calibration Matlab code

Camera and Projector Calibration
ProjMat.m
CameraParameters.m
FinalCalib.m

```
%*************************CALIBRATION*******************************
% In this algorithm the calibration of the camera and the projector is
% Calculated. The input data comes from the result generated from the
% corner detector algorithm. The output is the data related to the
% intrinsic and extrinsic values of both, the camera and the projector.

 %Camera Calibration:

load xycamera;
load XYZcamera;
 Ima=imread('CameraCalibration.JPG');
figure(1),imshow(Ima),hold on,title ('1. Camera Calibration Image')
figure(2),imshow(Ima),hold on,title ('2. Detected corners in the camera calibration image')
plot (xycamera(:,1),xycamera(:,2),'r+');hold off
figure(3),plot(xycamera(:,1),xycamera(:,2),'r+'), title ('2. Calibration in Camera Plane');
XYZcamera=XYZcamera%*10; %%Milimetros

%A is obtained from the matrices xy and XYC
%Construct A-matix

[Ac,Mc,mc] = ProjMat(xycamera, XYZcamera);

% Mc is m reshaped to a 3,4 Matrix
% Estimate the Camera Parameters
[fxc,fyc,oxc,oyc,Rc,Tc,alfac,omegac,kapac,PM]=CameraParameters(Mc);

% Final Calibration Matrix M

Mcamera = FinalCalib( fxc, fyc, oxc, oyc, Rc, Tc ); %PROJECTION MATRIX

fc=[fxc;fyc];      %CAMERA=L
cc=[oxc;oyc];      %CAMERA=L
%************************************************************
%                CAMERA RESULTS:                            *
%___________________________________________________________*
%a) Extrinsic Parameters of the Camera:                      *
%Rc;  %3X3 rotation Matrix                                   *
%Tc;  %Translation Vector                                    *
%alfac;                                                     *
%omegac;                                                    *
%kapac;                                                     *
%b) Instrinsic Parameters of the Camera:                    *
%oxc,oyc; %camera center                                    *
%fxc; %focal length in horizontal pixel size units          *
%fyc; %focal length in vertical pixel size units            *
%************************************************************
```

```
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\CameraCalibration.mat'
save(savefile, 'Rc','Tc','fc','cc');

%%%%_____%%%%

        %%%%PROJECTOR CALIBRATION%%%%%

load XYZprojector.mat; %cm
load xyprojector.mat;
Ima=imread('ProjectorCalibration.JPG');
figure(4),imshow(Ima);title('Projector Calibration Pattern');
figure(5),imshow(Ima);hold on;plot(xyprojector(:,1),xyprojector(:,2),'r+'),
title ('3. Detected corners in the projector calibration image');
hold off
figure(6), plot(xyprojector(:,1),xyprojector(:,2),'r+');
title ('4. Projector Calibration in Camera parameters')


%Construct A-matix

[Ap,Mp] = ProjMat(xyprojector, XYZprojector);

% Estimate Camera Parameters

[fxp,fyp,oxp,oyp,Rp,Tp,alfap,omegap,kapap]=CameraParameters(Mp);

%Final Calibration Matrix M

Mprojector = FinalCalib( fxp, fyp, oxp, oyp, Rp, Tp );
fp=[fxp;fyp];     %PROJECTOR=R
cp=[oxp;oyp];     %PROJECTOR=R

%*********************************************************
%            PROJECTOR RESULTS:                          *
%_____                                                   *
%a) Extrinsic Parameters of the Camera:                  *
%Rp;  %3X3 rotation Matrix                               *
%Tp; % %Translation Vector                               *
%alfap;                                                  *
%omegap;                                                 *
%kapap;                                                  *
%b) Instrinsic Parameters of the Projector:              *
%fxp; %focal length in horizontal pixel size units       *
%fyp; %focal length in vertical pixel size units         *
%*********************************************************
savefile='C:\Documents and Settings\Hector\Desktop\FINAL
ALGORITHM\parts\ProjectorCalibration.mat'
save(savefile, 'Rp','Tp','fp','cp');

% RELATING THE CAMERA AND THE PROJECTOR:

R=Rc*Rp';
T=Tp-Rp*Rc'*Tc


%SETTING NOTATION FOR NEXT STEP:
```

```
%R = RIGHT = PROJECTOR
%xR = xyprojector';
fR = [fxp;fyp];
cR = [oxp;oyp];
%L = LEFT = CAMERA
%xL = xycamera';
fL = [fxc;fyc];
cL = [oxc;oyc];
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\ProjCamaRel.mat'
save(savefile, 'R','T','fR','cR','fL','cL');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                END OF CALIBRATION                                   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

SUBRUTINES USED:

ProjMat.m

```
% This function finds the projection matrix. The inputs are the xy
% coordinates of the calibration pattern (Corners located in the
% picture) and the XYZ coordinates of the pattern (physical place where
% the corners in the pattern are located.

function [A,M1,m] =ProjMat ( xy, XYZ );

x = xy(:,1);
y = xy(:,2);
X = XYZ(:,1);
Y = XYZ(:,2);
Z = XYZ(:,3);
o = ones( size(x) );
z = zeros( size(x) );
Aoddrows = [ X Y Z o z z z z -x.*X -x.*Y -x.*Z -x ];
Aevenrows = [ z z z z X Y Z o -y.*X -y.*Y -y.*Z -y ];
A = [Aoddrows; Aevenrows];

%Calculate solution of A m = 0

[U, D, V] = svd( A );
m = V(:,end);

%Reshape m vector into M matrix

M1 = reshape(m,4,3)';
```

CameraParameters.m

```
% This function estimates the internal parameters of the camera.
% Such as focal length (fx,fy), Center point (ox,oy), Rotation
% Matrix (R), Translation Vector (T), alfa, omega and kappa
%(Translation angles) and the Projection matrix fixed with the % %
% correction factor (M)
```

```
function[fx,fy,ox,oy,R,T,alfa,omega,kapa,M]=CameraParameters(M1,inFront )
M=M1;
M = M/sqrt(M(3,1)^2+M(3,2)^2+M(3,3)^2);  %Denominator = scalar factor
s=-sign(M(3,4));
T(3)=s*M(3,4);
R=zeros(3,3);
R(3,:)=s*M(3,1:3);
q1=M(1,1:3)';
q2=M(2,1:3)';
q3=M(3,1:3)';
q4=M(1:3,4);
ox=q1'*q3;
oy=q2'*q3;
fx=sqrt( q1'*q1 - ox^2 );
fy=sqrt( q2'*q2 - oy^2 );
R(1,:)=s*(ox*M(3,1:3) - M(1,1:3) ) / fx;
R(2,:)=s*(oy*M(3,1:3) - M(2,1:3) ) / fy;
T(1)=s*(ox*M(3,4) - M(1,4) ) / fx;
T(2)=s*(oy*M(3,4) - M(2,4) ) / fy;
T=T';
[U,D,V]=svd(R);
R=U*V';
alfa=-atan(R(1,3)/R(3,3));
omega=asin(-R(2,3));
kapa=atan(R(2,1)/R(2,2));
```

FinalCalib.m

```
% This function reorders the projection matrix using all the different
% values obtained before.

function M=FinalCalib(fx,fy,ox,oy,R,T);

 M=[
    -fx*R(1,1)+ox*R(3,1) -fx*R(1,2)+ox*R(3,2) -fx*R(1,3)+ox*R(3,3) -fx*T(1)+ox*T(3);
    -fy*R(2,1)+oy*R(3,1) -fy*R(2,2)+oy*R(3,2) -fy*R(2,3)+oy*R(3,3) -fy*T(2)+oy*T(3);
    R(3,1)               R(3,2)               R(3,3)               T(3)     ];
```

Appendix F. Linear Ineterpolation Matlab code

Linearinterpol.m

```
%**********************LINEAR INTERPOLATION**********************
%This algorithm is used for linear interpolation.

function [resxN,resyN]=linearinterpol(matx,maty);
path(path,'C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts');
clear all
load cornersobject.mat;
matx=cornersplus(:,1);
maty=cornersplus(:,2);

[i,j]=size(matx);
N=1;
for a=1:i-1;
    if matx(a)>=matx(a+1) %&& matx(a)-matx(a+1)>~7;
       X1 = matx(a+1);
       Y1 = maty(a+1);
       X2 = matx(a);
       Y2 = maty(a);
    else
       X2 = matx(a+1);
       Y2 = maty(a+1);
       X1 = matx(a);
       Y1 = maty(a);
    End
X= X1:0.5:X2;
Y= Y1 + (Y2-Y1)/(X2-X1) * (X-X1);          %Linear Interpolation

    if a==1
       resxN=[X];
       resyN=[Y];
       N=N+1;
    end
    if a==2 && N==2;
       resxN=[resxN,X];
       resyN=[resyN,Y];
       N=N+1;
    end
    if a>=3
       resxN=1+[resxN-1,X];
       resyN=1+[resyN-1,Y];
       N=N+1;
    end
  end

linear=[resxN;resyN];
linear=linear'
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\linear.mat'
save(savefile, 'linear');
```

APPENDIX G. Relation between camera and projector

Centrofoto.m
Centrogrid.m

**********RELATION BETWEEN CAMERA AND PROJECTOR**********
%In this algorithm the relation between the picture, and the grid is %obtained

Centrofoto.m

```
%corner values related to image center:
%function [valorcentrofoto,i,j] = centrofoto (valorcorner,I);
path(path,'C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts');

load linear.mat
valorcorner=linear
CENTRO=size(I);
CENTRO=CENTRO'
i=CENTRO(1)/2; %pixels
j=CENTRO(2)/2; %pixels
XLPC=valorcorner(:,1);
YLPC=valorcorner(:,2);
Xdif=min(XLPC)-i;%i800
Ydif=max(YLPC)-j;%j600
CornDefCentX=XLPC-Xdif;
CornDefCentY=YLPC-Ydif;
ValCentCorners=[CornDefCentX,CornDefCentY]; %corner values in picture coordinate system... it is the
required for the faracross
figure(8), plot (CornDefCentX,CornDefCentY,'y+'), title('8. Corners related to camera world coordinate
system')
valorcentrofoto=ValCentCorners;
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\CentroPict.mat'
save(savefile, 'valorcentrofoto');
```

Centrogrid.m

```
%corner values related to grid center:
%function [valorgridcentro,CLX,CLY]=centrogrid(valorcorner);
path(path,'C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts');
clear
load linear.mat
valorcorner=linear
CLX=845; %pixels
CLY=521; %pixels
Xgp=valorcorner(:,1);
Ygp=valorcorner(:,2);
despX=min(Xgp)-CLX%800
despY=max(Ygp)-CLY%508
LinesDefCentX=Xgp-despX;
LinesDefCentY=Ygp-despY;
ValCent=[LinesDefCentX,LinesDefCentY];
valorgridcentro=ValCent
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\CentroGrid.mat'
```

APPENDIX H. TRIANGULATION

Triuangulation.m

```
%*****************************TRINGULATION*********************************
% In this algorithm the triangulation is performed.
% The input variables are the camera and projector calibration results, the linear interpolation and the
% relation between the camera and the  grid. The output file is the cloud of points of the object.

%function [Triangu]=tria2%(xc,cc,fc,xp,cp,fp,R,T)
path(path,'C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts');
load linear.mat
load CentroPict.mat
load ProjCamaRel.mat
load CentroGrid.mat
xp=valorgridcentro';
cc=cL;
fc=fR;
xc=valorcentrofoto';
cp=cR;
fp=fL;
N = size(xc,2);
kc=.015;
kp=1;

% normalize camera coordinates
xc2Bar = (xc - cc*ones(1,N)) ./ (fc*ones(1,N));
xcBar = comp_distortion(xc2Bar, kc);

% normalize projector coordinates
xpBar = (xp - cp *ones(1,N)) ./ (fp*ones(1,N));

% compute matrix A = [a b; c -1]
a =-(R(1,1)*xcBar(1,:) + R(1,2)*xcBar(2,:) + R(1,3)*ones(size(1,N)));
b = xpBar(1,:);
c =-(R(3,1)*xcBar(1,:) + R(3,2)*xcBar(2,:) + R(3,3)*ones(size(1,N)));

% compute Zc (depth of points in the camera frame of coordinates)
Zc = -(T(1)*ones(1,N) + T(3)*b) ./ (-a - b.*c);

% the cloud of 3D points
Triangu = (ones(3,1)*Zc) .* [xcBar; ones(1,N)];
Triangu = Triangu';
figure,plot(Triangu(:,1),Triangu(:,2),'.r'),title('Triangulation')
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\triangulation.mat'
save(savefile, 'Triangu');

a=Triangu(:,1);
b=Triangu(:,2);
c=Triangu(:,3);
one=min(a);
two=max(a);
dif1=one-two;
tree=min(b);
four=max(b);
dif2=tree-four;
```

```
five=min(c);
six=max(c);
dif3=five-six;
DIFERENCIAS=[dif1;dif2;dif3];
savefile='C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\3dcoord.mat'
save(savefile, 'DIFERENCIAS');
save('C:\Documents and Settings\Hector\Desktop\FINAL ALGORITHM\parts\teapot_ascii.xyz','Triangu','-
ASCII')
```