1 2 9 0

UNIVERSIDADE D
COIMBRA

Diogo Miguel Tavares Poço

VISUALCHAIN: VISUAL INSPECTION IN PRODUCTION LINES

Dissertation submitted to the Department of Electrical and
Computer Engineering of the Faculty of Science and Technology of
the University of Coimbra in partial fulfillment of the
requirements for the Degree of Master of Science

October 2020

VisualChain: Visual Inspection in Production Lines

Supervisor:

Prof. Paulo Jorge Carvalho Menezes

External Advisor:

Eng. Paulo Silva

Jury:

Prof. Jorge Manuel Moreira de Campos Pereira Batista
Prof. Rui Paulo Pinto da Rocha
Prof. Paulo Jorge Carvalho Menezes

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

October 2020

Agradecimentos

Começo por agradecer ao Professor Paulo Jorge Carvalho Menezes e ao Engenheiro Paulo Silva, junto com toda a equipa da SRAMPORT, por me terem dado a oportunidade de trabalhar neste projeto com todo o apoio necessário. Quero também agradecer à minha família por todo o apoio que me deram ao longo deste percurso. Aos meu colegas do MRL e IS3L que mantiveram sempre um ambiente animado e positivo mesmo durante os momentos mais difíceis, assim como ao Instituto de Sistemas e Robótica pelo espaço e material fornecido durante este tempo.

Obrigado a todos, são um grupo do qual me orgulho de fazer parte!

Coimbra, Outubro de 2020

Resumo

O controlo de qualidade é uma etapa necessária na linha de produção de todas as fábricas, o que se pode tornar num constrangimento se feito manualmente. Este procedimento manual não só pode ser lento, mas também pode estar sujeito a erros humanos. Este projeto faz um estudo de diferentes métodos para automatizar o procedimento de controle de qualidade em placas de correntes de bicicleta usando métodos não intrusivos por um meio de visão por computador. O objetivo é criar um sistema mais confiável e acelerar este processo de inspeção, exigindo o mínimo de interação com o operador. Para resolver este problema, três sistemas diferentes foram desenvolvidos. Um dos sistemas faz uma verificação baseada em imagens, usando apenas uma câmara e luz direcionada de vários ângulos para detetar e medir as placas considerando apenas duas dimensões, tornando-se assim um sistema rápido e preciso, mas sem efetuar uma verificação de qualidade completa. Os outros dois sistemas partilham o mesmo objetivo, complementar o sistema baseado em imagens, fazendo uma reconstrução tridimensional das placas. Estes são o Sistema de Varrimento por Laser e o Sistema de Luz Estruturada. As reconstruções resultantes destes dois sistemas são comparadas, discutindo assim o sistema mais apropriado. Após a aquisição da reconstrução das placas, a nuvem de pontos é comparada com uma outra nuvem de pontos modelo, completando o controle de qualidade com a verificação se a placa cumpre os requisitos necessários para o seu uso.

Palavras Chave: Reconstrução 3D, Metrologia, Sistema de Luz Estruturada, Varrimento por Laser, Métodos ópticos

Abstract

Quality control is a necessary step in the production lines of all factories, which can become a bottleneck if done manually. A manual quality control procedure not only is slower but can also be prone to human errors. Therefore, this project studies different methods to automate bicycle chainplates' quality control procedures using non-intrusive methods through computer vision. The goal is to create a more reliable system and accelerate the plates' inspection of an existing quality control, requiring minimal interaction with the operator. Three different systems were developed to solve this problem. An image-based quality control system uses only one camera and directed light to detect and measure the plates considering only two dimensions, making it a fast and accurate method. However, this system does not complete the plates' inspection, requiring another source of information. The other two systems share the same goal: complementing the image-based system by acquiring a three-dimensional reconstruction of the plates. These are the Laser Stripe Scanning and the Structured Light System. The resulted reconstructions of these two scanning systems are discussed, and the more efficient system is chosen. After acquiring the plates' 3D information, the point cloud is processed and compared to a model point cloud, completing the quality control by verifying if the plates have the needed requirements for their use.

Keywords: 3D-scanning, Metrology, Structured Light System, Optical methods

Contents

Acronyms

CCD Charge-coupled device. 2, 3

CMM Coordinate-Measuring Machine. 2

CT Computed Tomography. 3

DLT Direct Linear Transform. 10, 12

HSL Hue Saturation Luminance. 46

ICP Iterative Closest Point. 51

LIDAR Light Detection And Ranging. 2

LS Least-Squares. 28

LSS Laser Stripe Scanning. 3, 5, 36, 48–50, 55, 56

OCR Optical Character Recognition. 46

PCA Principal Component Analysis. 27–29, 51

PCA-RANSAC Random Sample Consensus applied to Principal Component Analysis. 27, 28, 50

RANSAC Random Sample Consensus. 27, 28

RGB Red Green Blue. 48

RMS Root Mean Square. 46, 49, 50

ROR Radius outlier removal. 31, 52

SLS Structured Light System. xv, 3, 5, 20, 21, 30, 51, 55, 56

SOR Statistical Outlier Removal. 30, 50

SVD Singular Value Decomposition. 12, 13

List of Figures

List of Tables

1 Introduction

Quality control is a process used in production lines to verify if a product has the needed requirements for its use. In the current days, every manufactured product has gone through a quality control procedure, that if inadequately executed, may result in severe consequences to the consumer, making it a crucial step in this industrialized world. Different quality control types may vary depending on the product, including chemical analysis, dimensional and weight measurements, color and texture verification, which are used on products like water, medicine, vehicles, and clothes. However, these inspections can consume a substantial amount of time, becoming a bottleneck in the production line.

This project studies and applies methods to improve both effectiveness and time performance in a dimensional quality control procedure. The inspection is executed on bicycle chains plates, making a series of measurements to verify if the plate is suitable for the end product. This work is done together with SRAMPORT.

SRAM is a bicycle component manufacturing known as a high-end cycling component brand, selling in markets globally. This high-status brand requires many branches, of which SRAMPORT is part. SRAMPORT's primary focus is producing high-quality bicycle chains, being the only factory producing bicycle chains for the SRAM company, which leaves it with significant responsibility to fulfill all their needs. This high demand leads to increased pressure on the production while, at the same time, the quality of the product has to be maintained.

1.1 Problem Statement

A bicycle chain is composed of inner and outer plates, rollers, and pins. However, in this project, the focus is only on the inner and outer plates. These are produced using a stamping method, in which a steel sheet is pressed by a stamping machine to produce the plates. This tool has four different modules, and each one prints a different inscription on the plates.

During this stamping process, the modules can be worn out by the use and, consequently, damaging the produced plates, such as deformations. One other cause of plate deformations is the metal sheet that can have imperfections as well. All these defects make the plates inadequate for the final product, so a quality control system is necessary to detect the faulty plates.

Having almost the whole production process automated, one of its few bottlenecks is detecting these faulty plates, which is done manually by an operator, making it time-consuming and prone to human errors.

1.2 State of the Art

In this current quality control procedure, the operator checks visually for defects on the plates' surface. He then uses a double cap to verify if the holes' diameter is between the minimum and maximum acceptable ranges, having the rest of the measurements checked with a touching probe, which has to be frequently calibrated. The efficiency of this process can be improved with the introduction of different systems. Here, only dimensional systems are studied due to the nature of this problem. There are various systems used to obtain spatial metrology, but this study only refers to systems used in small dimensions (millimeters), excluding systems like Light Detection And Ranging (LIDAR), which have an accuracy close to 1.5 cm [19].

One of the most used systems for this kind of metrology is the Coordinate-Measuring Machine (CMM) [41], which uses probes to acquire 3D points from the surface of an object. It moves in a three-dimensional Cartesian coordinate system with articulating heads, thus moving with a high degree of freedom. This system acquires a set of points one by one and uses them to calculate the geometric features [40][10], with the advantage of its high accuracy (in the order of dozens of micrometers [32]), at the cost of being a slow measuring method, due to its single point acquisition rate.

Another alternative is White Light Interferometry, an optical method that uses light wave properties to reconstruct objects. It uses the different phases between light waves, getting precise measurements in the vertical axis, having a resolution down to 0.3 nm [39]. Due to this, the system is sensitive to minor vibrations. The resolution of the orthogonal plan depends on the sensor used. When using a Charge-coupled device (CCD) camera, which is the most commonly used, the orthogonal resolution depends on the camera's resolution and the field of view [43].

The Computed Tomography (CT) is also a used system for dimensional metrology. It was initially developed for medical imaging but is now also used in industrial applications, where the reconstruction is acquired through X-rays taken from different angles [7]. This method allows taking measurements from the whole object surface in short periods. Current CT scanners have a vertical resolution of 0.5–0.625 mm, and a lateral resolution of approximately 0.5 mm [21].

All these three systems have a high price range. Therefore, simpler systems, like Laser Stripe Scanning, Photogrammetry, and Structured Light System, are prioritized.

The Laser Stripe Scanning (LSS) uses a stripe created by a dot laser scattered in a cylinder lens and captured by a CCD camera. The laser projection is viewed as a plane that intersects the object [47]. By using this intersection, a slice of the object is acquired [45]. The object moves in relation to the system, thus scanning its whole surface, reaching a vertical accuracy of 0.01 mm according to [48].

Photogrammetry is another used technique. Various photos of different profiles from a scene are taken, and their pixels are mapped between each capture [8]. By triangulating the mapped pixels, the 3D information can be acquired, obtaining the respective spatial representation, and reaching an accuracy of 0.2mm [22]. This technique is similar to a stereo system composed of two cameras. However, both these methods suffer from the matching problem. The matching problem occurs when mapping the 3D point to the corresponding pixels in the various frames.

The Structured Light System (SLS) is a solution for this matching problem. Instead of using a second camera, it uses a projector that projects patterns covering the whole scene, making a complete surface scan while the object is static, allowing a direct mapping between the projector and camera pixels. This system's resolution depends on the camera resolution and field of view, plus the projector resolution and projecting area. This system can reach a lateral resolution of 10 μm and a vertical resolution of 3 μm [37].

1.3 Objectives and Contributions

This project aims to study and apply techniques to develop an automated low-price quality control system that should replace the manual procedure currently used at the SRAMPORT production line. Upon completion, it should provide a faster, less tedious, and more reliable system to be applied by the quality control operator. For this, the Structured Light System and the Laser Stripe Scanning were chosen to be the research subject of this project, due to be the best cost-efficient methods for this problem. An image-based system was also developed, allowing a continuous verification of the plates.

The quality control has several parameters defined by SRAMPORT that must be verified after producing a plate. These are listed in the following table:

1.A	Plate's Height
1.B	Plate's Width
1.C	Plate's central Width
1.D	Hole's diameter
1.E	Holes' lateral decentralization
1.F	Holes' longitudinal decentralization
1.G	Holes' distance between each other
1.H	Readable Inscription
1.I	Plate's surface defects
1.J	Plate's thickness
1.K	Protuberance around the holes
1.L	Head chamfer
1.M	Chamfer undercut height
1.N	Central chamfer
1.O	Holes' interior chamfer
1.P	Plate's flatness

Table 1.1: Parameters analyzed in quality control

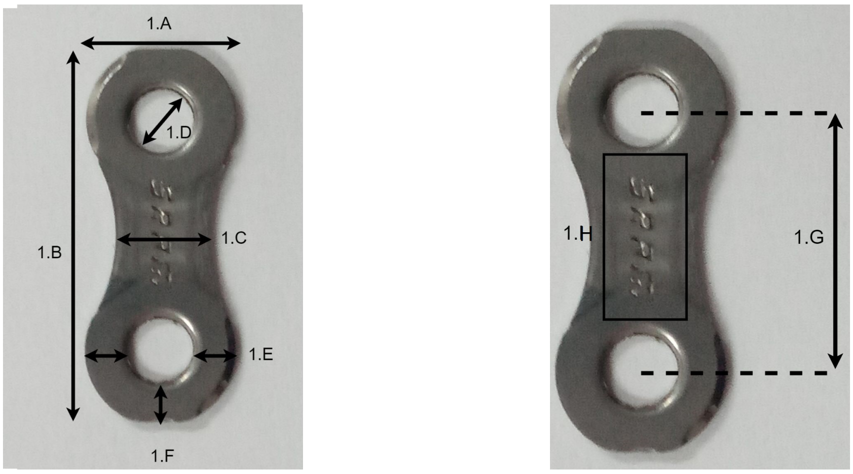


Figure 1.1: Quality Control using 2D coordinates

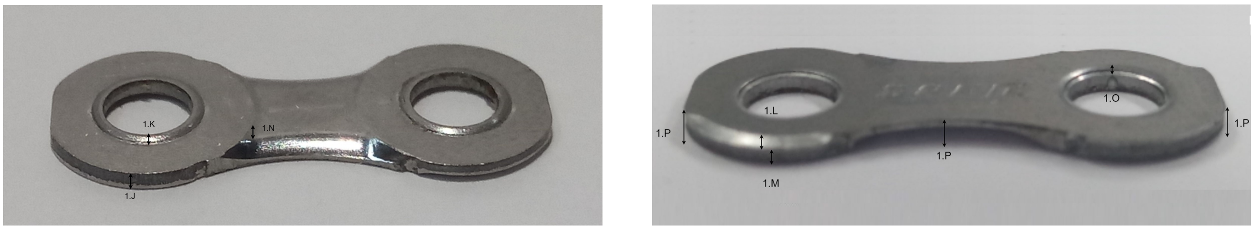


Figure 1.2: Quality Control using 3D coordinates

Note that the plate's measures cannot be displayed due to being confidential information. Where the plate's height is in the order of tens of millimeters. The system's maximum lateral error for the parameters 1.A to 1.G is 0.015mm, and the maximum vertical error for the parameters 1.J to 1.P is 0.03mm.

In the approach to this problem, these parameters are divided and inspected using two different systems. While the image-based system verifies the parameters 1.A to 1.H, the other parameters 1.I to 1.P need a 3D reconstruction to be analyzed, requiring the SLS or the LSS.

Figure 1.3: Examples of surface defects (1.I)

1.4 Outline of the Dissertation

This thesis is divided into five chapters. Chapter 2 delineates the fundamentals of the chosen methods for each system. Chapter 3 describes the setups used, with a study about the systems' characteristics. In chapter 4, a description of the procedure is done, together with the results obtained. Chapter 5 finalizes the thesis with the conclusions obtained during this work.

2 Fundamentals

All systems developed in this project use a camera as a source of information. This chapter describes the fundamentals of the used methods in the acquisition and processing of this information.

2.1 Camera Model

The pinhole camera model is based on cameras made of a box that blocks all the light except for a small aperture (pinhole) that allows the light to travel through it into the box without suffering any internal reflections when obtaining the projection made from the incident light. A representation of this camera is shown in the figure 2.1.

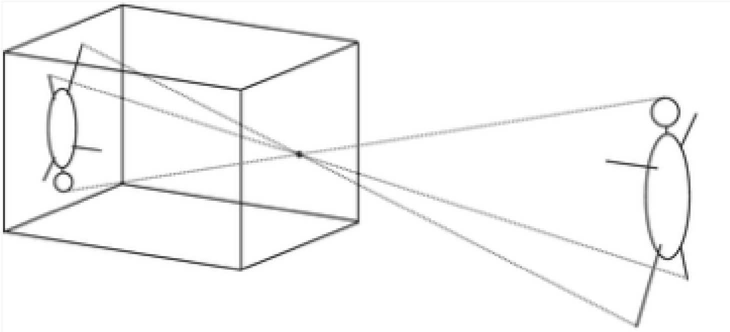


Figure 2.1: Pinhole Camera Representation [35]

The model establishes the relationship between a point in the 3D world and its projection on the 2D image plane. This relationship is given by:

$$\begin{aligned} x &= f\frac{X}{Z} \\ y &= f\frac{Y}{Z} \end{aligned} \tag{2.1}$$

where (x,y) are image coordinates, (X,Y,Z) are the corresponding world coordinates, and f is the focal length of the lens, representing the distance between the center of projection and the image plane. This equation (2.1) assumes that the camera position is in the origin of the world reference with its optical axis along the Z-axis. The image coordinates are then converted to frame coordinates (u,v), representing the coordinates in pixels of the frame, each pixel having the information of an area in the image plane.

$$\begin{aligned} x &= -s_x(u - u_0) \\ y &= -s_y(v - v_0) \end{aligned} \tag{2.2}$$

where s_x and s_y are the pixel size (in mm), and u_0 and v_0 are the coordinates correspondent to the center of the image (in pixels). Combining the equations (2.1) with (2.2), it is possible to transform the world coordinates to frame coordinates using the following equations:

$$\begin{aligned} u &= -\frac{f}{s_x Z} X + u_o \\ v &= -\frac{f}{s_y Z} Y + v_o \end{aligned} \tag{2.3}$$

Which can be represented in its matrix form as:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} -\frac{f}{s_x} & 0 & u_0 \\ 0 & -\frac{f}{s_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix},$$

Intrinsic and Extrinsic Parameters

However, these equations do not apply when the camera's referential is not the same as the world's referential. Assuming a rotation and translation between these two, the relation between a point in world coordinates and camera coordinates is given by:

$$\bar{\mathbf{M}} = \mathbf{R}\mathbf{M} + \mathbf{t}$$

This way, the point $\mathbf{M}$ from the world coordinate system can be converted to the camera coordinate system, resulting in $\bar{\mathbf{M}}$. This is achieved by using the rotation matrix, $\mathbf{R_{3x3}}$, and the translation vector, $\mathbf{t_{3x1}}$, that finds the location of the camera coordinate system with respect to the world coordinate system, creating the extrinsic matrix, $\begin{bmatrix}\mathbf{R},\mathbf{t}\end{bmatrix}$. Having the relationship between the camera coordinate system and the world coordinate system given by the extrinsic matrix, the intrinsic matrix, $\mathbf{K}$, gives the relationship between the frame coordinates and the camera coordinates.

$$\mathbf{K} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

with (u_0, v_0) being the coordinates correspondent to the center of the image plane, α and β the scale factors in the image ($\alpha = -\frac{f}{s_x}$ and $\beta = -\frac{f}{s_y}$) and γ the parameter represents the skew coefficient of the two axes u and v in the frame. Knowing the intrinsic and extrinsic matrices, the relationship between a 3D world point $\mathbf{M} = \begin{bmatrix}X, Y, Z\end{bmatrix}^T$ and its corresponding frame coordinate $\mathbf{m} = \begin{bmatrix}u, v\end{bmatrix}^T$ can be obtained by:

$$s\tilde{\mathbf{m}} = \mathbf{K}\begin{bmatrix}\mathbf{r1} & \mathbf{r2} & \mathbf{r3} & \mathbf{t}\end{bmatrix}\tilde{\mathbf{M}}, \tag{2.4}$$

note that the points $\tilde{\mathbf{M}}$ and $\tilde{\mathbf{m}}$ are the homogeneous coordinates of $\mathbf{M}$ and $\mathbf{m}$, with s being a scale factor. So the point $\mathbf{m}$ is related with $\mathbf{M}$ by the projection matrix $\mathbf{P}$:

$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \begin{bmatrix} p1 & p2 & p3 & p4 \\ p5 & p6 & p7 & p8 \\ p9 & p10 & p11 & p12 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \mathbf{p}_3^T \end{bmatrix}. \tag{2.5}$$

Radial Distortion

Most cameras suffer from some radial distortion due to the lens used, causing a displacement in the pixels' position. As shown in the figure 2.2, this radial distortion can be categorized into Barrel distortion (b) or Pincushion distortion (c).

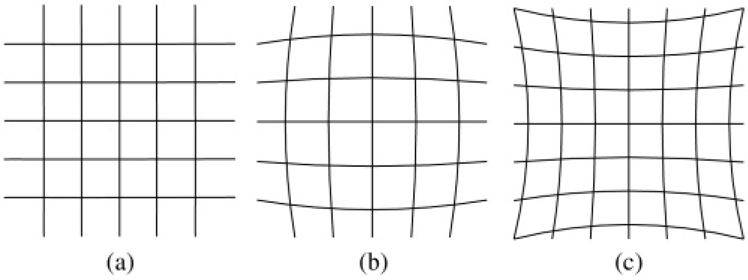


Figure 2.2: Types of radial distortion [16]

There are various models used to estimate the distortion in order to remove or alleviate it. The most commonly used model characterizes it using the following polynomial equation [23]:

$$F(r) = rf(r) = r(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 + \cdots) \tag{2.6}$$

with $k_1, k_2...$ being the distortion coefficients and $r^2 = x^2 + y^2$. These distortion coefficients can acquired during the camera calibration, through the camera's intrinsic parameters.

2.2 Camera Calibration

The camera calibration has three essential goals: obtaining the camera's intrinsic parameters, extrinsic parameters, and distortion coefficients, which can be done by acquiring the camera's projection matrix. This section describes two different calibration methods.

2.2.1 Direct Linear Transform

Direct Linear Transform (DLT) is a method used to determine the intrinsic and extrinsic matrices when given a set of observed points in the world and their corresponding image coordinates [3]. This method has 11 degrees of freedom from the 11 parameters, five intrinsic, and six extrinsic. So the DLT method requires at least 6 points to solve the unknown variables since each point results in two linear equations. The more points used, the more accurate the results are.

This linear approach considers that $\tilde{\mathbf{m}}$ and $\mathbf{P}\tilde{\mathbf{M}}$ are vectors with the same direction.

$$\tilde{\mathbf{m}} \times \mathbf{P}\tilde{\mathbf{M}} = 0 \tag{2.7}$$

Where the cross product of $\tilde{\mathbf{m}} \times \mathbf{P}\tilde{\mathbf{M}}$ can be replaced by the matrix multiplication of $[\tilde{\mathbf{m}}_X]\mathbf{P}\tilde{\mathbf{M}}$, being $[\tilde{\mathbf{m}}_X]$ the Skew-symmetric matrix of $\tilde{\mathbf{m}}$:

$$[\tilde{\mathbf{m}}_X] = \begin{bmatrix} 0 & -\tilde{m}_3 & \tilde{m}_2 \\ \tilde{m}_3 & 0 & -\tilde{m}_1 \\ -\tilde{m}_2 & \tilde{m}_1 & 0 \end{bmatrix}.$$

So, by transforming the dot product into the previous matrix multiplication, the equation

(2.7) can be expressed as:

$$\begin{bmatrix} 0 & -1 & v \\ 1 & 0 & -u \\ -v & u & 0 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{M}}^T & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} \\ \mathbf{0}_{1\times4} & \tilde{\mathbf{M}}^T & \mathbf{0}_{1\times4} \\ \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \tilde{\mathbf{M}}^T \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \tag{2.8}$$

Where $\mathbf{0}_{1\times4} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$. By applying equation (2.8) to every observed point, it is possible to stack these linear equations together and use them to solve the projection matrix unknowns, resulting in:

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 - u_1Z_1 - u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 - v_1Z_1 - v_1 \\ & & & & & \dots & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -u_nX_n & -u_nY_n - u_nZ_n - u_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_nX_n & -v_nY_n - v_nZ_n - v_n \end{bmatrix} \begin{bmatrix} p1 \\ p2 \\ p3 \\ p4 \\ p5 \\ p6 \\ p7 \\ p8 \\ p9 \\ p10 \\ p11 \\ p12 \end{bmatrix} = 0, \tag{2.9}$$

with n being the number of observed points. Equation (2.9) can also be written as $\mathbf{Lp} = 0$, where the matrix $\mathbf{L}$ contains the linear equations. Using the Singular Value Decomposition (SVD), the values of $\mathbf{p}$ can be acquired by decomposing $\mathbf{L}$ into:

$$\mathbf{L} = \mathbf{U\Sigma V}^T$$

where $\mathbf{\Sigma}$ is a diagonal matrix containing the singular values and $\mathbf{U}$, and $\mathbf{V}$ are unitary matrices. $\mathbf{V}$ has the singular vectors corresponding to the singular values of $\mathbf{\Sigma}$. By using the vector of $\mathbf{V}$ that corresponds to the smallest singular value in $\mathbf{\Sigma}$, $\mathbf{v}_{12}$, it is possible to obtain the values of $\mathbf{p}$ that minimizes the error, being $\mathbf{p} = \mathbf{v}_{12}$. After acquiring the matrix $\mathbf{P}$, it is possible to obtain the values of $\mathbf{K}$, $\mathbf{R}$ and $\mathbf{t}$ by assuming that:

$$\mathbf{P} = \mathbf{KR}\begin{bmatrix} I_3 | \mathbf{C} \end{bmatrix} = \begin{bmatrix} \mathbf{A} | \mathbf{a} \end{bmatrix}, \text{ with: } \mathbf{A} = \begin{bmatrix} p1 & p2 & p3 \\ p5 & p6 & p7 \\ p9 & p10 & p11 \end{bmatrix} \text{ and } a = \begin{bmatrix} p4 \\ p8 \\ p12 \end{bmatrix},$$

with $\mathbf{C}$ being the translation between the world referential and the camera referential. Knowing this, $\mathbf{t}$, $\mathbf{K}$, and $\mathbf{R}$ are computed by:

$$\mathbf{t} = \mathbf{A}^{-1}\mathbf{a}$$

Since $\mathbf{K}$ is a triangular matrix, it is possible to use the QR decomposition to obtain $\mathbf{K}$ and $\mathbf{R}$. Knowing that $\mathbf{A}^{-1} = \mathbf{R}^T\mathbf{K}^{-1}$, the QR decomposition of $\mathbf{A}^{-1}$ is:

$$\mathbf{A^{-1}} = \mathbf{QR}$$

,

where $\mathbf{R}^T = \mathbf{Q}$ and $\mathbf{K}^{-1} = R$.

2.2.2 Zhang's Calibration Method

Zhang's calibration method is a more commonly used alternative to the DLT process due to its self-calibration techniques and accurate results [49], being the chosen calibration method in this thesis.

Intrinsic and Extrinsic Parameters

This method uses various planes as the calibration objects. The use of planes allows the Z-axis of the world coordinate system to be parallel to the plane's normal, which results in the Z coordinate of every point being 0. With this, the equation 2.4 can be written as:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{r1} & \mathbf{r2} & \mathbf{t} \end{bmatrix}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{2.10}$$

Making the point $\mathbf{M}$ related to $\mathbf{m}$ by a homography $\mathbf{H}$:

$$\tilde{\mathbf{m}} = \mathbf{H}\tilde{\mathbf{M}} \tag{2.11}$$

with $\mathbf{H} = \begin{bmatrix} \mathbf{h1} & \mathbf{h2} & \mathbf{h3} \end{bmatrix} = \lambda\mathbf{K}\begin{bmatrix} \mathbf{r1} & \mathbf{r2} & \mathbf{t} \end{bmatrix}$. The matrix $\mathbf{H}$ is acquired using the same method as the DLT, by obtaining the linear equations for each point and using the SVD to calculate the unknowns. Unlike matrix **P**, the matrix **H** does not have any information about the vector $\mathbf{r}_3$, which means that the matrix **R** cannot be extracted from **H**. Due to this, the QR decomposition cannot be used to obtain **K**, so another alternative is used to exploit the orthonormal constraints of **R**. Since **r1** and **r2** are orthonormal, the following constraints are true:

- $\mathbf{r_1}^T\mathbf{r_2} = 0,$
- $|| \mathbf{r_1} ||=|| \mathbf{r_2} ||= 1.$

Knowing that $\mathbf{r}_1 = \mathbf{K}^{-1}\mathbf{h}_1$ and $\mathbf{r}_2 = \mathbf{K}^{-1}\mathbf{h}_2$, the constraints can be written as:

- $\mathbf{h}_1^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_2 = 0,$
- $\mathbf{h}_1^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_1 = \mathbf{h}_2^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{h}_2,$

with $\mathbf{K}^{-T}\mathbf{K}^{-1}$ being a symmetric matrix. Let:

$$\mathbf{B} = \mathbf{K}^{-T}\mathbf{K}^{-1} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{12} & B_{22} & B_{23} \\ B_{13} & B_{23} & B_{33} \end{bmatrix},$$

by defining a vector **b** as $\mathbf{b} = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T$ and creating a vector $\mathbf{v}_{ij}$ such that: $\mathbf{h}_i^T\mathbf{B}\mathbf{h}_j = \mathbf{v}_{ij}^T\mathbf{b}$. It is possible to write the constraints like:

- $\mathbf{v}_{12}^T\mathbf{b} = 0,$
- $\mathbf{v}_{11}^T\mathbf{b} - \mathbf{v}_{22}^T\mathbf{b} = 0.$

With this, it is possible to create a matrix **V**, such that $\mathbf{Vb} = 0$, by stacking the constraints of each image, making it a 2n x 6 matrix, where n is the number of images with the plane in a different position relative to the camera, resulting in:

$$\begin{bmatrix} \mathbf{v}_{12}^T \\ (\mathbf{v}_{11} - \mathbf{v}_{22})^T \\ \cdots \end{bmatrix} \mathbf{b} = \mathbf{0}.$$

Since **b** has 6 variables, this method needs at least 3 images to obtain the **b** values. The values are calculated by decomposing **V** using the SVD and acquiring the vector corresponding to the smallest singular value. After reconstructing the matrix **B**, it is possible to obtain the intrinsic parameters **K** through the Cholesky decomposition, since **K** is symmetric. Having the intrinsic parameters **K**, the extrinsic parameters can be obtained from (2.10):

- $\mathbf{r}_1 = \lambda \mathbf{K}^{-1}\mathbf{h}_1$,
- $\mathbf{r}_2 = \lambda \mathbf{K}^{-1}\mathbf{h}_2$,
- $\mathbf{r}_3 = \mathbf{r}_2 \times \mathbf{r}_1$,
- $\mathbf{t} = \lambda \mathbf{K}^{-1}\mathbf{h}_3$,

with the scale factor $\lambda = \frac{1}{\|\mathbf{K}^{-1}\mathbf{h}_1\|} = \frac{1}{\|\mathbf{K}^{-1}\mathbf{h}_2\|}$. This solution can be refined using a maximum likelihood estimation. Assuming that an identically distributed noise corrupts every image's points, it is possible to minimize the following function with the Levenberg-Marquardt algorithm,

$$\sum_{i=1}^{n}\sum_{j=1}^{m} \| \; \mathbf{m_{ij}} - \hat{\mathbf{m}}(\mathbf{K}, \mathbf{r_i}, \mathbf{t_i}, \mathbf{M_j}) \; \|^2, \tag{2.12}$$

where $\hat{\mathbf{m}}(\mathbf{K}, \mathbf{r_i}, \mathbf{t_i}, \mathbf{M_j})$ corresponds to the projection of point $\mathbf{M_j}$'s estimation in image i, according equation (2.10) and $\mathbf{m_{ij}}$ is its position on the image plane.

Distortion Coefficients

Now that the intrinsic parameters are known and assuming that the camera suffers from either a barrel distortion or pincushion distortion, it is possible to estimate the radial distortion using the following equations:

$$\check{\mathrm{u}} = u + (u - u_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2 + k_3(x^2 + y^2)^4], \tag{2.13}$$

$$\check{\mathrm{v}} = v + (v - v_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2 + k_3(x^2 + y^2)^4], \tag{2.14}$$

where the coordinates (u,v) correspond to the ideal coordinates in the frame, (ŭ, v̌) to the real frame coordinates with radial distortion, and (x,y) to the normalized image coordinates. By stacking these two equations of every point of every image, it is possible to obtain the following matrix and vectors:

$$\begin{bmatrix} (u-u_0)\left(x^2+y^2\right) & (u-u_0)\left(x^2+y^2\right)^2 & (u-u_0)(x^2+y^2)^4 \\ (v-v_0)\left(x^2+y^2\right) & (v-v_0)\left(x^2+y^2\right)^2 & (v-v_0)\left(x^2+y^2\right)^4 \\ \dots & & \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} = \begin{bmatrix} \tilde{u}-u \\ \tilde{v}-v \\ \dots \end{bmatrix}. \tag{2.15}$$

If (2.15) is written as $\mathbf{Dk} = \mathbf{d}$, the values of the coefficients $\mathbf{k}$ are given by the linear least-squares solution: $\mathbf{k} = \left(\mathbf{D}^T\mathbf{D}\right)^{-1}\mathbf{D}^T\mathbf{d}$. Once again, this result can be refine by a maximum likelihood estimation, adding the distortion coefficients:

$$\sum_{i=1}^{n}\sum_{j=1}^{m} \|\ \mathbf{m_{ij}} - \breve{\mathbf{m}}(\mathbf{K}, \mathbf{k_1}, \mathbf{k_2}, \mathbf{k_3}, \mathbf{R_i}, \mathbf{t_i}, \mathbf{M_j})\ \|^2\ . \tag{2.16}$$

2.3 Image-based System

The Image-based system captures images of the plates, with its camera axis being perpendicular to the plane that supports the plates. Due to the plates' small height, a projection of the plates to the reference plane is made, thus assuming their height to be zero. With this, the system calculates the distances based only on two dimensions. As shown in the Fig. 2.3, the further a point is from the camera's optical axis, the larger the error. In the figure, the obtained coordinates are shown as red while the real coordinates are blue. To alleviate the error when calibrating, the reference plane was considered at the plates' top surface.

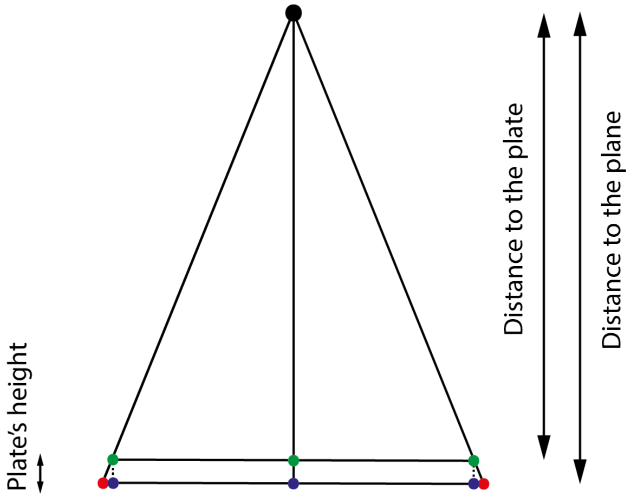


Figure 2.3: Image-based System's error representation, the acquired coordinates do not correspond to the reality.

The image-based system starts to segment the captured image by extracting the pixels corresponding to the plates. Once those pixels are known, the selection of the points used to calculate the specific distances in the metrology is executed (the selection of these points is shown in the chapter 4.1.3). In the calibration procedure, the reference plane's translation and rotation to the camera are obtained and the homography between these two calculated. The frame coordinates are converted to coordinates in the reference plane through the homography, using them to measure the pretended distances. During the selection of points, the system detects the area where the inscription is and throughout lighting from various angles, it tries to segmentate the characters. The following study compares the tested segmentation techniques used to extract the plates from the captured image and for the character segmentation in the inscription.

Image Segmentation Study

There are various image segmentation techniques, and each one can be used in different situations. In [31], they study various image thresholding methods on document images, sharing similarities to the plates' detection due to the high contrast with the background [2]. Between the various threshold methods used, there are a few that can be grouped in different categories.

The methods of Otsu and Kittler [31] are clustering-based methods. Otsu minimizes the weighted sum of variances of the foreground and background pixels to establish the threshold, while Kittler uses the variance in Gaussian density functions to have a distribution of background and foreground pixels.

Kapur [31], Sahoo [31], Yen [46], and Li [20] are entropy-based methods, where the cross entropy of the distribution of gray levels is used to calculate the threshold.

Sauvola, White and Bernsen [31] are locally adaptive methods, where they adapt the threshold value of each pixel to the local image characteristics. Sauvola method uses the local variance while White and Bernsen use the local contrast.

The various segmentation techniques were tested, having the best results with the Li cross entropy-based threshold for this application. However, the previous methods do not work with the inscription detection due to the similarity between the characters and the plate. Therefore, different methods, like Gabor filters and the Sobel filter, were used to solve this problem.

Gabor filters are linear filters based on Gabor functions used as texture discrimination[12].

Its convolution kernel is a product of a Gaussian and a cosine function [15], and acts as a local band-pass filter with specific optimal joint localization properties in the spatial and frequency domain [9], resulting in local texture segregation. An algorithm based on Gabor filters typically uses various kernels, each analyzing the frequency content in an image in specific directions.

The Sobel filter is often used to detect edges in images, using the difference between pixel values in the image [33]. It uses two masks to calculate a derivative approximation for both x and y axes from the frame. The Sobel operators are [38]:

$$\mathbf{G_x} = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad \mathbf{G_y} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

The resulted derivatives from these two masks are combined into a single matrix by: $\mathbf{G} = \sqrt{\mathbf{G_x}^2 + \mathbf{G_y}^2}$.

In this project, the Sobel Operators were chosen to segment the inscription. The masks were applied in captures using lighting from different angles, increasing the gradient in the pretended areas. Instead of using the matrix $\mathbf{G}$ for the characters segmentation, a threshold is applied to both derivative matrices individually, obtaining more consistent results.

2.4 Laser Stripe System

The Laser stripe system uses a laser to project a red line across the scene and captures it with a camera. Due to the camera placement and rotation to the laser, the stripe should appear nearly horizontal in the camera's frame, maximizing its precision [26], resulting in every column in the image capturing it.

2.4.1 Stripe Segmentation

This stripe's segmentation aims to extract the laser from the image. It converts the stripe position to the camera's coordinates frame by acquiring the peak value in every column of the captured image, corresponding to the center of the stripe. With this, the system requires that the highest light intensity captured by the camera comes from the laser projection and not other light sources.

When using this method, the accuracy is limited to the image's resolution, thus to refine the results, a sub-pixel algorithm is used. D. K. Naido [26] conducted a comparative analysis between various sub-pixel algorithms such as Gaussian approximation, Center of Mass, Linear Interpolation, Parabolic Estimator, and Blais and Rioux Detectors. In the end, Blais and Rioux Detectors with eighth order linear filters gave higher accuracy when compared to the other algorithms, which made a suitable candidate for this project. The Blais and Rioux algorithm with eight order linear filter is:

$$\delta = \frac{g(v)}{g(v) - g(v+1)} \tag{2.17}$$

where g(v) is:

$$g(v) = f(v-4)+f(v-3)+f(v-2)+f(v-1)-f(v+1)-f(v+2)-f(v+3)-f(v+4) \tag{2.18}$$

The stripe's peak is situated at a sub-pixel coordinate $(v + \delta)$. The method uses the intensity values from the surrounding neighbors of the peak coordinate v, where f() corresponds to the intensity of those neighbors. By acquiring the coordinates correspondent to the laser, both the laser calibration and the reconstruction are now possible.

2.4.2 Laser Calibration

In the laser calibration, the main goal is to obtain the position of the laser plane (π_L) with respect to the camera, where the laser plane is formed between the laser's light source and the projected line. This plane is later used to triangulate an object's 3D coordinates by its intersection with the scanning object's surface.

The laser is projected in various target planes π_C, as shown in figure 2.4, where the intersection between π_L and π_C results in the laser line L_C. When using Zhang's technique to calibrate the camera, the rotation and translation $[\mathbf{R}|\mathbf{t}]$ of the target planes can be acquired.

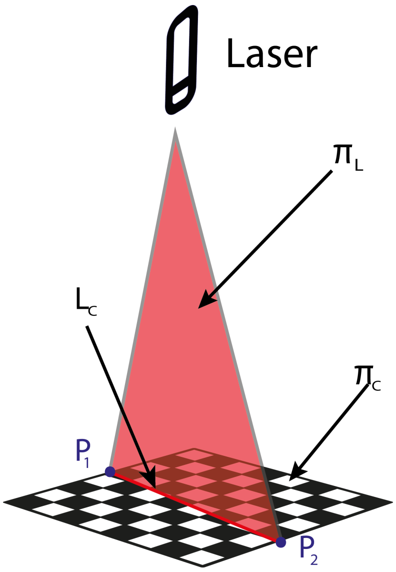


Figure 2.4: Calculation of the laser plane

To obtain the direction vector of L_C, two points belonging to the line are needed. First, the coordinates of those two points in the camera reference frame are acquired by the intersection between the laser line and the target plane's outline. These points are then converted to world coordinates using a homography to the calibration plane, resulting in $\mathbf{P_1}$ and $\mathbf{P_2}$. Once the points $\mathbf{P_1}$ and $\mathbf{P_2}$ of each plane are known, the normalized vector that connects them, $\mathbf{D_L} = \begin{bmatrix} m & n & l \end{bmatrix}^T$ is calculated, being the vector that represents the intersection of the laser projection with the calibration plane in world coordinates.

Let $\mathbf{D}$ be a matrix with all the vectors $\mathbf{D_L}$ stacked together, and $\mathbf{n}_L$ the normal vector of the laser plane. It is known that $\mathbf{n}_L$ is perpendicular to the laser plane, and therefore perpendicular to the vectors $\mathbf{D_L}$, hence the dot product of $\mathbf{n}_L$ and the vectors $\mathbf{D_L}$ is 0. So, it is possible to calculate $\mathbf{n}_L$ by finding a vector that minimizes its dot product with all the vectors $\mathbf{D_L}$ [18].

$$\mathbf{D} \cdot \mathbf{n}_L = 0 \leftrightarrow \begin{bmatrix} m_1 & n_1 & l_1 \\ m_2 & n_2 & l_2 \\ \vdots & \vdots & \vdots \\ m_n & n_n & l_n \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0$$

Since the solution to this problem is the right null space of the matrix $\mathbf{D}$. The matrix $\mathbf{D}$ can be decomposed by singular value decomposition, resulting in:

$$\mathbf{D} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T,$$

obtaining the coordinates of $\mathbf{n_L}$ from the last column of $\mathbf{V}$, $\mathbf{v_3}$, allowing to calculate the plane equation, which is given by:

$$ax + by + cz + d = 0,$$

with (a,b,c) being n_l coordinates and d the minimization of the square sum distances between all points in each laser line and the laser plane.

$$d = \frac{\sum_{i=0}^{n}[(am_i + bn_i + cl_i)s_i^2 + (ax_{1i} + by_{1i} + cz_{1i})s_i]}{\sum_{i=0}^{n} s_i} \tag{2.19}$$

Where (m_i, n_i, l_i) are the vector coordinates of a vector i in $\mathbf{D}$, (x_{1i}, y_{1i}, z_{1i}) are the point coordinates of $\mathbf{P_1}$ corresponding to the vector i and s_i is the distance between the two points $\mathbf{P_1}$ and $\mathbf{P_2}$.

2.5 Structured light system

The Structured Light System is similar to a stereo system composed of two cameras. In a two-camera stereo system, each camera captures a frame of a scene and the system triangulates the pixels corresponding to the shared scene points in each image. In an SLS stereo system, instead of a second camera, it has a projector that projects one or more patterns identifying its frame pixels. The triangulation is also done between the two frames, but the matching is obtained with the camera's captured frame and the frame projected by the projector, as shown in the figure 2.5.

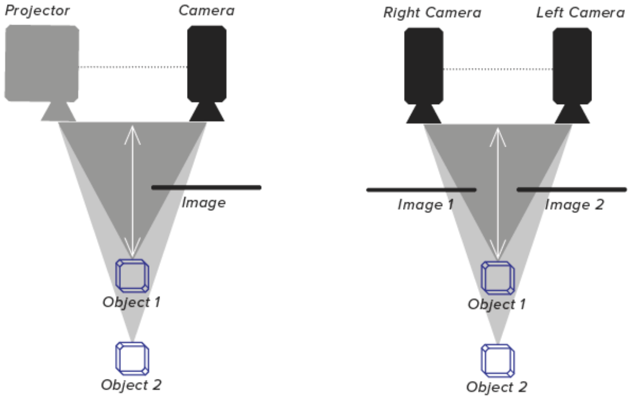


Figure 2.5: Left: SLS **Right:** Stereo system of two cameras [1]

2.5.1 Stereo Systems

With the SLS and the two-camera stereo system being similar, they share the same triangulation principle. This section describes the triangulation using two cameras for simplicity, which also applies to SLS.

Triangulation

When capturing a frame of a scene by a single camera, it is not enough to recover 3D information of the captured points. Instead, it is only possible to trace the rays that intersect the center of the camera and each scene point. So, one way to have the 3D position of the scene points is by inserting a second camera, creating a stereo system. Since the two cameras' centers and a random scene point always form a plane, it is possible to calculate the 3D coordinates of the scene point with respect to the stereo system by intersecting the two rays from the cameras' centers to the scene point.

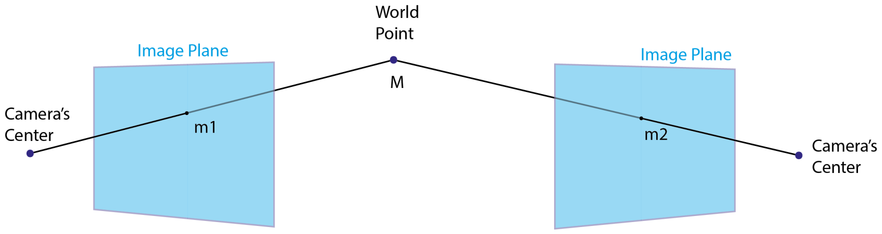


Figure 2.6: Representation of stereo system triangulation

As shown in the figure 2.7, at least two frames are required to obtain the $\mathbf{M}$ point coordinates. The Projection Matrix $\mathbf{P}$ gives the relation between the point $\mathbf{M}$ and the corresponding pixel in the camera's frame from the equations (2.5) and (2.4) [24]. Since the two vectors, $\tilde{\mathbf{m}}$ and $\tilde{\mathbf{M}}$ have the same direction as shown in the equation (2.7), the cross product between them is 0.

$$\tilde{\mathbf{m}} \times \mathbf{P}\tilde{\mathbf{M}} = 0,$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \mathbf{p}_3^T \end{bmatrix} \tilde{\mathbf{M}} = \begin{bmatrix} v\mathbf{p}_3^T - \mathbf{p}_2^T \\ \mathbf{p}_1^T - u\mathbf{p}_3^T \\ u\mathbf{p}_2^T - v\mathbf{p}_1^T \end{bmatrix} \tilde{\mathbf{M}} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.20}$$

.

The third line from the result of the cross product is the combination of the first two lines, not giving any additional information, making it possible to create a system of linear equations using only the first two lines of the cross product. By adding the second camera, the coordinates of $\tilde{\mathbf{M}}$ can be calculated using the following linear equations:

$$\begin{bmatrix} v\mathbf{p}_3^T - \mathbf{p}_2^T \\ \mathbf{p}_1^T - u\mathbf{p}_3^T \\ v2\mathbf{p2}_3^T - \mathbf{p2}_2^T \\ \mathbf{p2}_1^T - u2\mathbf{p2}_3^T \end{bmatrix} \mathbf{X} = 0, \tag{2.21}$$

where (u2,v2) are the coordinates corresponding to the same point, but in the second camera view, and **P2** is the second camera projection matrix. Using the singular value decomposition to solve the equation, the 3D homogeneous coordinates of the point **M** are acquired.

Projection Matrices Acquisition

The point **M** can be triangulated once the cameras' projection matrices are acquired. Assuming that $\mathbf{M_1}$ and $\mathbf{M_2}$ are the corresponding points to **M** in their cameras' coordinate system:

$$\begin{aligned} \mathbf{M_1} &= \mathbf{R_1}\mathbf{M} + \mathbf{T_1}, \\ \mathbf{M_2} &= \mathbf{R_2}\mathbf{M} + \mathbf{T_2}, \end{aligned} \tag{2.22}$$

where $\mathbf{R_1}$, $\mathbf{T_1}$ and $\mathbf{R_2}$, $\mathbf{T_2}$ are the rotation matrices and translation vectors that link the world coordinates system with the cameras' coordinate system, extracted from extrinsic parameters. If the matrix **R** and vector **T** are the translation and rotation between the two cameras, the points $\mathbf{M_1}$ and $\mathbf{M_2}$ can be written as [50]:

$$\mathbf{M_1} = \mathbf{R}^T(\mathbf{M_2} - \mathbf{T}) \tag{2.23}$$

From the equations 2.22 and 2.23 , **R** and **T** can be written as:

$$\begin{aligned} \mathbf{R} &= \mathbf{R_2}\mathbf{R_1^T} \\ \mathbf{T} &= \mathbf{T_2} - \mathbf{R}\mathbf{T_1} \end{aligned} \tag{2.24}$$

Knowing the rotation and translation between the two cameras, the projection matrices can be calculated. The first camera is considered the center of the coordinate system, having the projection matrix:

$$\mathbf{P_1} = \mathbf{K_1}[I|0] \tag{2.25}$$

While projection matrix from the second camera is:

$$\mathbf{P_2} = \mathbf{K2}[R|T] \tag{2.26}$$

In the used structured light system, instead of having a second camera, it has a projector. Since the projector projects a frame instead of capturing it, its frame coordinates are obtained using other methods where the projector is seen as an inverse camera.

2.5.2 Types of Coded Structured Light

A few different strategies are used when it comes to projecting a pattern in a structured light system. They have in common the use of codewords to assign the pixels in the camera to the projector's frame, which results in every pixel that captures a pattern, having its codeword. Different techniques can be categorized into direct coding, spatial neighborhood, and time-multiplexing [30], depending on the system's priority: accuracy or acquisition time.

In the direct coding techniques, there are grey level coding and color-coding. The grey level coding uses various pixel intensities to code every line of pixels, while the color-coding uses different hue values instead. Both of these algorithms require two projection patterns to code the vertical and horizontal lines. The low amount of pattern projections make these techniques time-efficient, but with high noise sensitivity. Wust and Capson [42] changed the color technique in a way that it requires only one pattern projection, which makes it possible to be used on moving surfaces with the requirement of the scanned surfaces being color neutral and without large discontinuities.

The spatial neighborhood techniques are mainly used for a continuous acquisition, due to the majority of the algorithms only requiring one projection pattern. These techniques label the codeword of a point based on the position of its neighbor points. Some of these patterns of points can be generated based on the *De Bruijin sequence* [30] obtained from different *De Bruijin graphs* or based on M-arrays, where the projection is split into sub-matrices, each one containing a different pattern, as shown in the figure. Some M-arrays algorithms also use colors to facilitate pattern segmentation.

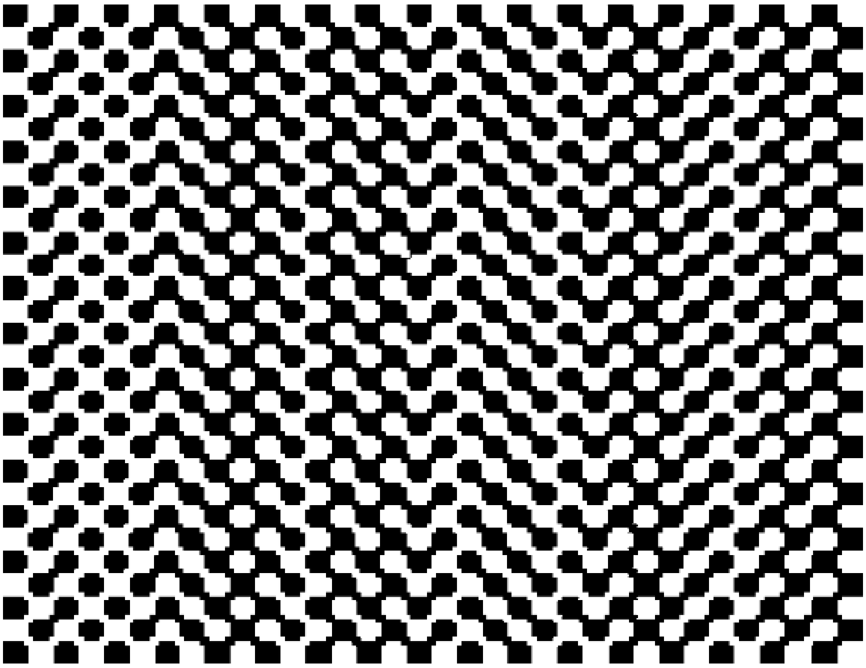


Figure 2.7: *De Bruijin sequence* [30]

The time multiplexing techniques usually require various amounts of projections to obtain the reconstruction. One of these methods is the binary coded projection that only uses two intensity levels, a high and a low value, resulting in white and black pixels. The pattern of stripes can be encoded by binary code or Gray code. The Gray coded pattern makes the system more robust to errors than the binary, due to each codeword only having a hamming distance of 1 bit to each of its neighbors' codewords. It also reduces the number of stripes resulting in fewer segmentation errors. This method can obtain accurate results at the cost of time, due to the projection of many patterns where the object has to be completely static. N-ary codes method is another time multiplexing technique similar to the binary coded projection, using more intensity levels, keeping the high contrast between them.

2.5.3 Structured Light Code Segmentation

The chosen technique was the binary coded pattern with Gray codification, where the pixels only have two intensities, which leads to labeling each pixel as: on, off, or uncertain. The classic segmentation methods for this technique usually use two projections to acquire the pixels' labeling limits, one with all the pixels at a high intensity (on) and the other at low (off). The pixels are identified as uncertain if the difference between the high and low values is lower than a predefined threshold. During the pattern projections, the rest of the pixels are labeled based on their intensity value.

In this project, a more accurate segmentation method is used, where the direct and global light is computed and used to classify all pixels.

Direct and Global Lighting

The direct and global lighting values are acquired from a sequence of comparisons between captured frames of the different projected patterns. The pixel values captured by each frame are split into two components, the direct and the global light [27]. The direct light is reflected from the projection directly into the camera, while the global light is the remaining light reflected from other surfaces. Considering L as the radiance from a certain point to the camera, L_d as the direct light component, and L_g as the global light component, let:

$$L = L_d + L_g \tag{2.27}$$

If a scene has points lit by the projector (L^+) and points non-lit (L^-). Let the radiance of these points be expressed as:

$$\begin{aligned} L^+ &= L_d + \alpha L_g \\ L^- &= (1-\alpha)L_g \end{aligned} \tag{2.28}$$

Where α is the fraction of lighted pixels on the projection, L^+ is composed of the direct component from the lighted pixels and the α fraction of the global component that would have received if the scene was fully lit by all the projection pixels. Considering that L^-, the non-lit radiance of the same point, is acquired by the compliment of the pattern used to obtain L^+, it only has the global component $(1-\alpha)$ since there is no direct light lighting the point.

The problem with these equations is that they assume that the deactivated pixels in the projection do not generate any light, which is not the case. So, considering a variable b that represents the fraction of light produced by a deactivate pixel compared to an activated pixel, the equation (2.28) can be written as:

$$\begin{aligned} L^+ &= L_d + \alpha L_g + b(1-\alpha)L_g \\ L^- &= bL_d + (1-\alpha)L_g + \alpha b L_g \end{aligned} \tag{2.29}$$

These equations take into account the extra direct and global light produced by the deactivated pixels. The pattern used to light the scene in this project is a set of stripes coded with the Gray code, which means that in each projected image, half of the pixels are lit, resulting in $\alpha = 0.5$. It is possible to calculate the value of L_d and L_g of a point, knowing L^+ and L^-:

$$L_p^+ = \underset{0<i\leq k}{max} I_i(p), \qquad L_p^- = \underset{0<i\leq k}{min} I_i(p) \tag{2.30}$$

Where L_p^+ and L_p^- are the lit and non-lit light captured by the camera's pixel location p, which is obtained by the maximum and minimum intensity value $I_i(p)$ of all images i. From (2.29) with $\alpha = 0.5$, $L_d(p)$ and $L_g(p)$ can be computed by:

$$L_d(p) = \frac{L_p^+ - L_p^-}{1-b}, \qquad L_g(p) = 2\frac{L_p^- - bL_p^+}{1-b^2}. \tag{2.31}$$

Pixel Classification

Having the direct and global light values, it is now possible to classify each pixel in the camera frame, into on, off, or uncertain, using Yi Xu classification rules [44]:

$L_d < m \rightarrow$ pixel is uncertain
$L_p < min(L_d, L_g) \rightarrow$ pixel is off
$L_p > max(L_d, L_g) \rightarrow$ pixel is on
otherwise $\rightarrow$ pixel is uncertain

Where L_p is the pixel intensity, and m is a predefined value. The advantage of this method compared to using a simple threshold is the detection of uncertain pixels during the classification. This classification is applied for every projected pattern, but the Yi Xu rules can go even further and give better results when used with the inverse pattern, for every pattern projected.

$L_d < m \rightarrow$ pixel is uncertain
$L_d > L_g$ & $L_p > L_{\bar{p}} \rightarrow$ pixel is on
$L_d > L_g$ & $L_p < L_{\bar{p}} \rightarrow$ pixel is off
$L_p < L_d$ & $L_{\bar{p}} > L_g \rightarrow$ pixel is off
$L_p > L_g$ & $L_{\bar{p}} < L_d \rightarrow$ pixel is on
otherwise $\rightarrow$ pixel is uncertain

Where $L_{\bar{p}}$ is the pixel intensity in the frame with the inverse projected pattern. This second method can give a better classification at the cost of doubling the captured frames.

Pixel Coding

Non-uncertain pixels are classified for each projected pattern, resulting in the creation of codewords. Each codeword is formed by a bit sequence of 0's and 1's, corresponding to each pixel's classification in all acquisitions. If a pixel is on, the bit is asserted to 1. If a pixel is off, the bit is asserted to 0. The complete codeword is formed by two halves, one representing the horizontal sequence and one representing the vertical, where the least significant bit of each half corresponds to the latest projected pattern. The codewords' length is the same as the number of patterns projected, which itself is limited by the projector's resolution. The best precision is reached when using $ceiling(log_2(max(Projector'sHeight, Projector'sWidth)))$ gray code iterations, allowing to each projector's pixel have a unique codeword. Once the codeword is complete, each half is converted to decimal resulting in the (u,v) projector coordinates. The gray code is converted to decimal code using the following algorithm:

```
Set G = Gray code
Set D = 0
While G differs from 0:
    D = G XOR D
    G = G » 1
```

After obtaining the decimal equivalent of each pixel's codeword, the relation between the projector's and camera's pixels is known for each acquisition.

2.5.4 Reference Plane Removal

With the pixel matching between the camera's and projector's frames, it is possible to triangulate and reconstruct the scanned object's three-dimensional model. With the reconstruction obtained, there is an excess of information in the point cloud, which in this case, is the reference plane, requiring its removal. This process removes all the plane points from the reconstruction, resulting in a point cloud of the isolated plate. This is accomplished by obtaining the plane equation of the reference plane, which is acquired by scanning it during the calibration procedure. The used method to obtain this equation was the Random Sample Consensus (RANSAC) of a Principal Component Analysis (PCA) (PCA-RANSAC), since methods like the Least-Squares (LS) method highly depends on the plane's coordinate system [25].

Principal Component Analysis

The PCA method is applied to the matrix containing all the point cloud points of the reference plane. Through a statistic method, PCA finds the best fitting plane based on the direction and variances of the point cloud [34]. First, the point cloud is centered at the origin, and its covariance matrix is computed. The eigenvectors and eigenvalues can be obtained from the covariance matrix and used to calculate the plane's normal vector. The plane's normal vector is the eigenvector with the lowest eigenvalue, since it is the vector with the smallest amount of variance from the point cloud [28]. The point cloud can be centered by obtaining the plane's centroid and subtracting it to all the points.

Considering $\mathbf{X}$ to be the point cloud of the plane, of size n, and $\bar{\mathbf{X}}$ the same size matrix with the centroid coordinates in all rows.

$$\mathbf{B} = \mathbf{X} - \bar{\mathbf{X}} \tag{2.32}$$

Now the covariance matrix can be calculated by:

$$\mathbf{\Sigma} = \mathbf{B}^T\mathbf{B} \tag{2.33}$$

Having the covariance matrix, it's eigenvalues and eigenvectors are determined such that:

$$\mathbf{\Sigma x} = \lambda \mathbf{x} \tag{2.34}$$

Where λ correspond to the eigenvalues. Using the lowest eigenvalue, the eigenvector equivalent to the plane normal is obtained.

PCA-RANSAC

The RANSAC is an iterative method used to reduce the impact of the invalid data points (outliers) in the plane's normal calculation [11]. Rather than using the whole point cloud to calculate the plane, it uses small portions of it and compares them. In this project, the RANSAC algorithm is combined with the PCA. The PCA-RANSAC algorithm requires as input parameter the point cloud, the number of points in the portion of the point cloud used in each iteration, the threshold value that determines if a point is an inlier or outlier, and the number of inliers required to consider that the model succeed.

The algorithm starts by randomly selecting a portion of the point cloud points. Using the PCA method, the plane equation is obtained using that portion. Each point is verified and classified as an inlier or outlier using the plane equation, based on the predetermined threshold. If the model has the number of inliers required to succeed, the algorithm outputs the model. Otherwise, it repeats the process. The process is repeated until it has more inliers than the threshold or until the maximum iterations are overpast, resulting in a failed model.

2.6 Point Cloud Analysis

Now that the point cloud of the isolated plate was acquired, this information needs to be analyzed. Firstly, the noise needs to be removed. This process is done by detecting and eliminating outliers using the Statistical Outliers Removal algorithm [4]. Once the point cloud is clean of noise, a comparison between this and a pre-existing model point cloud of the plate is made. This comparison is accomplished by calculating every vector of the nearest point of the reconstructed point cloud to the model and verifying if the height between these two is within the defined threshold (process described in the section 4.4). To find the nearest point between the two point clouds, the KD Tree method is used, alleviating the computation time required for this process. After acquiring the set of points corresponding to the defects, the Radius Outlier Removal algorithm is applied, removing any isolated point. This chapter gives a description about KD Trees, Statistical Outliers Removal and Radius outlier removal.

2.6.1 KD Tree

The KD Tree is a data structure method which is used for associative searches [5]. In this project, the KD Tree is used to find the nearest point to another given point without requiring high computational time. This method structures the data using nodes distributed between various levels, where each node has two pointers, LOSON and HISON, that either are null or point to another node in the next level of the tree. These pointers are used based on a discriminator, with the number of the discriminators being the same as the number of dimensions, which in this case are the axis X, Y, Z. These discriminators are shared with other nodes on the same level and vary throughout the levels, but always maintaining the same sequence. Being used to select the path to the next node in the tree. Assuming an existing node $\mathbf{P_1}$ in a level where the discriminator is the coordinate X, with the pointers LOSON($\mathbf{P_1}$) and HISON($\mathbf{P_1}$). When introducing a second node $\mathbf{P_2}$, if the X value of $\mathbf{P_2}$ is lower than $\mathbf{P_1}$, then the pointer LOSON($\mathbf{P_1}$) points to $\mathbf{P_2}$, if higher the pointer HISON($\mathbf{P_1}$) points to $\mathbf{P_2}$. In case the discriminator is equal, the process repeats using the next discriminator. Once the data is structured, the nearest point can easily be obtained by operating through the nodes.

2.6.2 Statistical Outliers Removal

The Statistical Outlier Removal (SOR) algorithm is a commonly used method to remove these noisy points (outliers) from a point cloud [29]. The statistical outlier removal method requires as input parameters the point cloud, the number of neighbors used to compare, and a multiplier to identify outlier points [4]. The algorithm starts to calculate the average Euclidean distance d_i to the selected number of nearest neighbors, using the KD Tree method. Then, it calculates the global mean distance μ and the standard deviation ξ:

$$\mu = \sum_{i}^{N} \frac{d_i}{N} \tag{2.35}$$

$$\xi = \sqrt{\frac{1}{N} \sum_{i}^{N} (d_i - \mu)^2} \tag{2.36}$$

Where N is the number of points. The points are classified as inliers or outliers using the standard deviation, the global mean distance, and the average distance of each point to its neighbors. Each point is an outlier if: $d_i >= \mu + \alpha\xi$, where α is the predetermined multiplier factor.

Example

For this example, an SLS reconstruction of a medal i1s made. This medal has low reflective properties on its top but highly reflective on its sides, making it hard to adjust the light settings for the scan and, consequently, create unwanted points. The result of the reconstruction is shown on the left of figure 2.8. As shown in the figure, the point cloud quality is not desirable, so to solve this problem, the SOR filter is applied. Once the SOR filter is applied, and the unwanted points removed, the clean point cloud on the right of figure 2.8 is obtained.

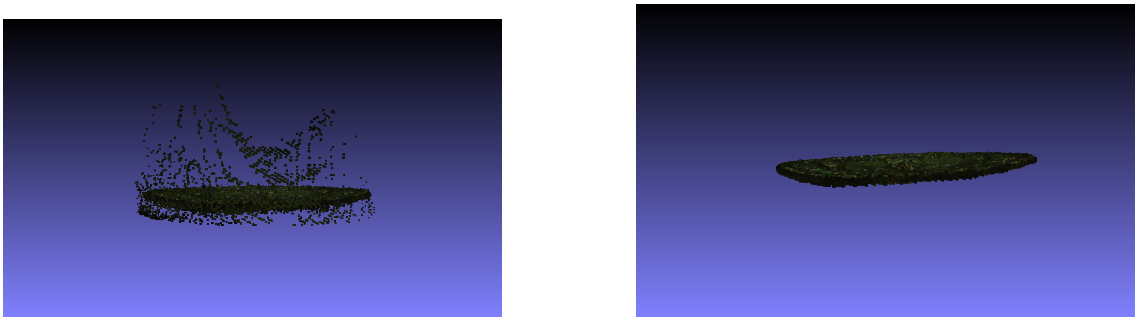


Figure 2.8: Left: Medal reconstruction before applying the SOR filter; **Right:** Medal reconstruction after applying the SOR filter

2.6.3 Radius Outlier Removal

Like the Statistical outlier removal, the Radius outlier removal (ROR) is a denoising filter. However, instead of calculating the threshold distance with the mean distance and the standard deviation, it uses an already given distance as the threshold. The distance of each point to a given number of the nearest neighbors is calculated using the KD Tree. The point is rejected if any of the distances to its neighbors is higher than the given threshold. This algorithm is applied iteratively to the defected points, resulting in clusters that highlight the defective zone.

3 Setup

This chapter describes the systems' materials, position and function in the setup. It also contains a study on the LSS's limitations and ways to maximize its accuracy.

3.1 Image-based System

The system consists of a camera vertically pointing to the plane area where the plate is analyzed (this will be called reference plane). The camera used is the Microsoft LifeCam Studio, with a resolution of 1920x1080 pixels. The reference plane is made by a translucent material, sitting above a reflective cylinder followed by five LEDs. The light from the five LEDs is scattered by the cylinder and diffused by the translucent plane, allowing a uniform illumination of the reference plane without lighting the plates directly. This makes the plates' segmentation possible without occurring any reflection errors. This setup also contains four other LED lights surrounding the plane where the quality control occurs, with each LED pointing directly to the reference plane and separated with an equal distance to the camera and a 45º degree angle between each other. Unlike the mechanism used to illuminate the reference plane, these four LEDs use the reflections to highlight the plates' inscription and surface scratches. The camera captures four frames, each with only one LED active, acquiring the reflections from different angles. To avoid interferences from any outside illumination, the whole system is surrounded by a dome blocking all direct light. All LEDs are connected to a microcontroller Arduino Uno that synchronizes each LED activation with the program using serial communication.

In this system, the accuracy is highly dependent on the camera distance to the reference plane and its resolution. The distance between the reference plane and the camera has to be high enough for the operator to place the plates under it but cannot be too high that the acquisition quality starts to suffer. Since the plates' height is minimal compared to this distance, the system does not suffer from any surface out of focus.

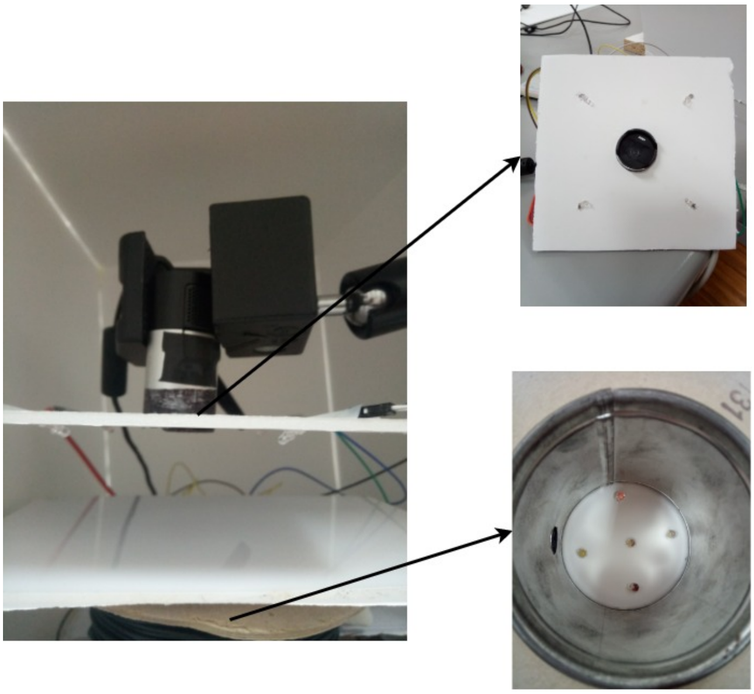


Figure 3.1: Two dimensional system composed of one camera and LEDs

3.2 Laser Stripe System

The Laser stripe system consists of a laser projecting a stripe onto a plane and captured by a camera. The plates sit on this plane, which itself is a moving platform. The platform moves a predefined distance, then the camera captures a frame of the laser stripe, and this process repeats until the whole plate is scanned. This means that the plate moves a constant distance between each frame captured by the camera. The smaller the movement between each frame, the more data is acquired from the object surface. The moving platform's direction is perpendicular to the laser projection line, with any variance directly impacting the systems reconstruction results. This movement is controlled by a step motor, which itself is controlled by a microcontroller. The laser and the camera in this system are static, with the only moving objects being the platform, and consequently, the plates.

In the Laser stripe system, the resolution of the point cloud depends on two factors. When projecting the laser stripe on the plate's surface, a slice of the object is acquired. The points belonging to this slice are two dimensional, and the resolution of this slice reconstruction depends on the camera and laser. The laser's position and focal length have a direct impact on the stripe's width. In the ideal system, the line would have the smallest possible width that the camera could still capture, reducing the segmentation errors when obtaining the stripe's center coordinates. However, the reconstruction quality also depends on the camera's position and resolution. The closer the camera is to the laser plane, and the higher the camera's resolution, the higher the density and accuracy of the points acquisition. The camera position is dependent on its focal length, with a smaller the focal length allowing a shorter distance to the laser plane. The third dimension of the reconstruction is introduced with the movement of the platform. For each translation done by the platform, the corresponding translation is applied to the slice acquired. The precision of this dimension is highly dependent on the precision of the traveled distance.

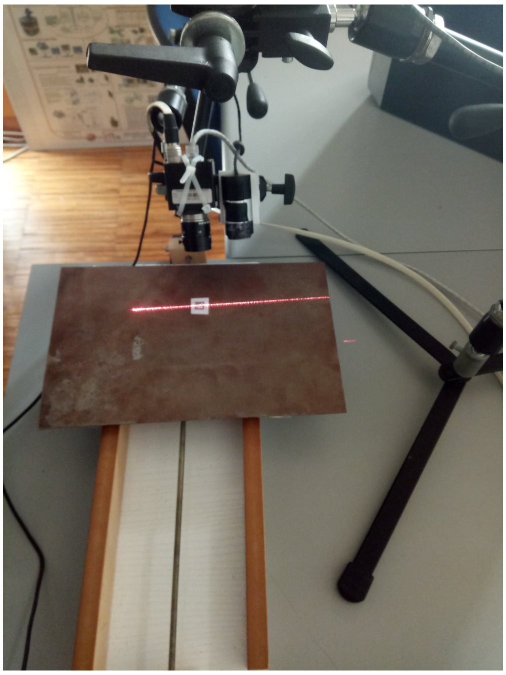


Figure 3.2: Laser Stripe Scanning composed of a camera and a laser.

In this study, due to the erratic movement of the platform, an extra step is added. An Aruco marker was attached to the plane and used to track its position alongside the scanning process. This may restrict the camera position and focal length, due to the need of capturing the marker in every shot. The laser stripe system's advantage is the focus area being on the laser plane instead of the whole reference plane, which is lost when implementing the marker. This system was tested using two cameras, the Microsoft LifeCam Studio using a resolution of 1920x1080 pixels, and the Allied Vision Prosilica GC650C, with 640x480 pixels. The laser used was a Lasiris Inc SNF-501L with a wavelength of 685nm.

3.3 Structured Light System

The structured light system setup consists of a projector pointing to the scene where the plates are scanned and a camera capturing the projector's projected pattern. This setup is surrounded by a dome covering the whole system, blocking any undesired light. Contrary to the laser system, the object is now static, maintaining its initial position during the full scan. The accuracy of the system depends on both the camera and the projector. A higher and more accurate point cloud is obtained with a higher camera's and projector's resolution and a closer distance to the reference plane, which is limited by their focal length.

Since the reconstruction is acquired by triangulating the camera's pixels with the projector's pixels, both should have a similar resolution, distance to the reference plane, and field of view. A significant difference in these can result in errors. For instance, a much higher camera resolution with similar distance and field of view as the projector could result in the same projector's pixel by various cameras pixels, resulting in reconstruction errors. Ideally, the frame captured by the camera should differentiate every single pixel of the projector without capturing the same pixel twice. The camera's exposure time should also be higher than the projectors refresh rate to avoid captures of frames not completely refreshed. The camera used in this system was the Allied Vision Prosilica GC650C, with a resoltution of 640x480 pixels. The projector used was the Hitachi CP-RX70, with a resolution of 1024x768 pixels and a refresh rate of 80Hz. However, the projector's lens did not allow the image's focus for the pretended distance, with the camera only capturing a fraction of the projection.

Figure 3.3: Structured Light System composed of a camera and a projector.

3.4 Setup Characteristics

In this section, the influence of the angle between the camera and the laser is studied, using the LSS. The α angle is obtained between the laser projection and the camera's optical axis.

From this angle, it is possible to determine the β angle from the laser plane's normal vector and the camera's optical axis, with the value of $|90° - \alpha|$, as it is shown in fig 3.4. The closer β is to zero, the more accurate the reconstruction is. But a high α value may also result in some limitations.

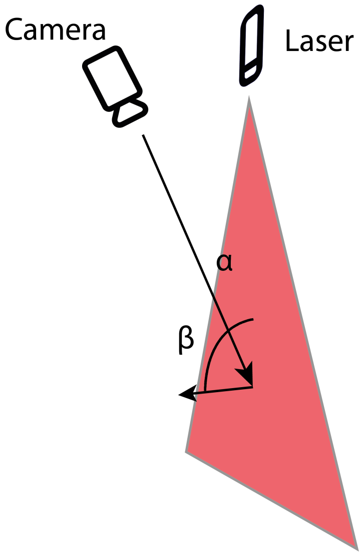


Figure 3.4: α and β angles

3.4.1 Occlusion Points

The occlusion points are inevitable when using the Laser Stripe System. They occur when there's a point in space that cannot be reached by either the laser plane or the camera view, which is highly dependent on the angle between these two. This is mainly caused by the object's height, surface's form, and rotation towards the camera and laser, so there is a need to adapt the system according to the scanned object. The addition of more cameras or lasers could be one solution to lower the occlusion points, but that would increase the system's cost.

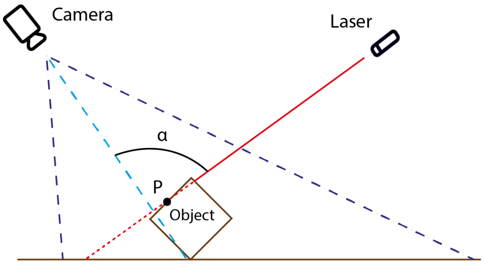


Figure 3.5: Representation of occlusion points

In figure 3.5, the system suffers an occlusion when trying to scan the point P. The dark blue dashed lines represent the camera view with the light blue dashed line representing the camera's optical axis, and the red line the laser projection. In this situation, the laser projection cannot reach the chosen point, which means that the laser planes need to be adjusted. This can be done in two ways, rotating the laser alone, resulting in a lower α, or rotating both the camera and the laser to preserve the α angle.

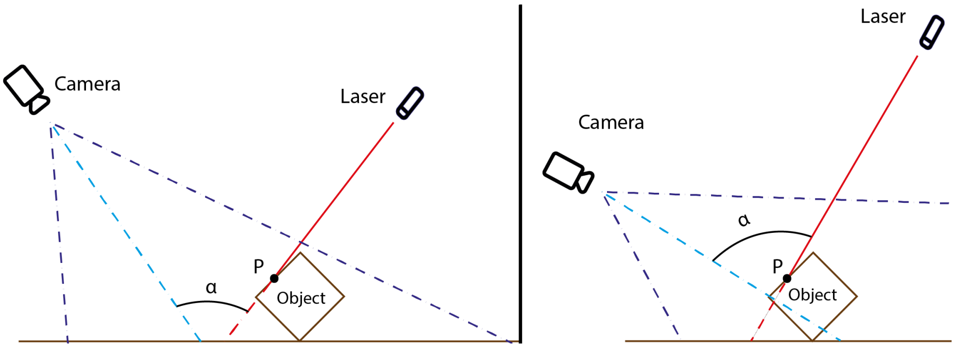


Figure 3.6: Setup changes. **Left**: Laser rotation changed, α value lowered when compared to previous system. **Right**: Both camera and laser position changed, equal α value compared to previous system

In this example, none of the setups would scan the whole top surface, needing to adapt the camera position and rotation as well. This could also be solved by rotating the object, due to its setting not being ideal for the scan.

3.4.2 Saturation Problems

The saturation problems are caused when the sum of the reflected light from a surface saturates the camera's sensor due to its high intensity, creating errors in its corresponding pixels and their neighbors. This problem mainly happens when the laser hits a highly reflective material, resulting in less scattering of the light with a more directional reflection. The light intensity peaks when the angle of the reflected ray with the surface's normal vector is equal to the angle of the incident ray, as shown in fig 3.7.

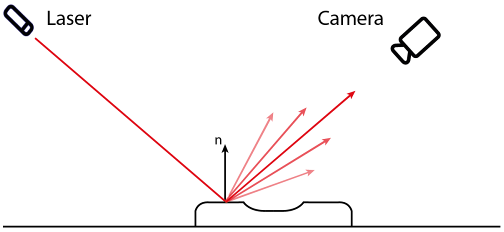


Figure 3.7: Laser reflection on a glossy surface

The following simulation neglects this problem, with the camera not suffering from any saturation. However, when applying to real hardware, this problem could restrict some angles between the camera and the laser when structuring the system.

3.4.3 Simulation

In this experiment, the Laser Stripe System's setup uses some functionalities of Blender modeling and ray tracer software, exploring shader programming possibilities, and thus eliminating errors derived from the equipment and lighting. This setup consists of a laser pointing vertically to a reference plane and a camera capturing the laser projection using different camera angles to study the impact of the angle β on the system's accuracy. In the simulation, the object movement relative to the system does not suffer any errors with a translation of the same exact distance between every frame.

To demonstrate the previous section 3.4.1 about occlusion points, the object in the figure 3.8 was generated.

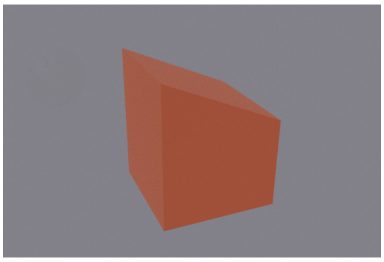
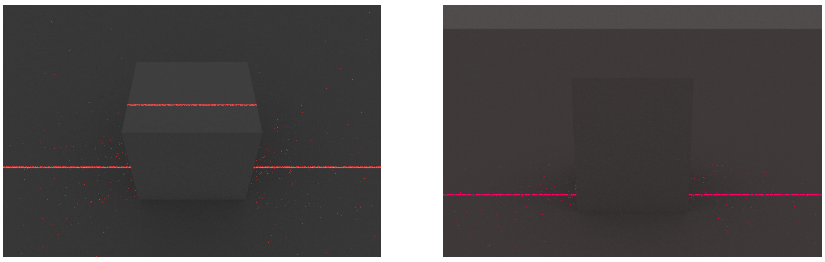


Figure 3.8: Simulation of an object in blender

There were used two cameras with two different positions. They are at the same distance to the point in the plane intersected by the optical axis but with a different α. It is noticeable that in figure 3.9.(a) there are no major occlusions, contrary to figure 3.9.(b), where the occlusions do not allow a proper scan of the object. Meaning that the setup (a) is better for this situation.

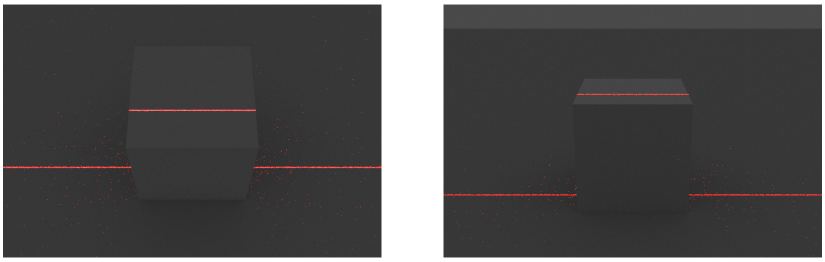


(a) $\alpha = 26^{\circ}$

(b) $\alpha = 63^{\circ}$

Figure 3.9: System simulation using two different camera positioning

At the same time, if the object is a cube, the setup (b) gives better results than the setup (a). This is due to the β angle in the setup (b) being lower than in the setup (a), which results in the same height being represented with more pixels in the captured frame, as shown in the fig 3.10.



(a) $\alpha = 26^{\circ}$ **(b)** $\alpha = 63^{\circ}$

Figure 3.10: System simulation using two different camera positioning

There were used two cameras with a resolution of 600x400 and a cube with a size of 10cm. After acquiring the reconstruction, the average error in the measured height was 0.0055cm in the setup (a) and 0.0014cm in the setup (b). The average error in the measured width was 0.0418cm in (a) and 0.03730cm in (b). Note that these are simulation values, not suffering from any errors derived from the stripe segmentation or equipment limitations, such as focal lens. The laser stripe has a uniform intensity in the whole projection without suffering from any scattering light.

4 Methodology

In this chapter, a description of the executed procedures in the developed systems is presented. At the end of each system's section, the results acquired by each system are displayed.

4.1 Image-based system

In this system, the verified parameters of the quality control are 1.A to 1.I from the table 1.1. The parameters are represented by the fig 1.1, where the system measures the plates only assuming two dimensions.

4.1.1 Calibration

The camera intrinsic and extrinsic parameters were acquired using Zhang's Calibration method, referred to in the section 2.2.2. The plane used in this process had a chessboard pattern printed in it due to its easy corner detection using the Harris operator [17], followed by the Forstner method [13] to refine the corner location to a subpixel value. The more corners used in the chessboard pattern, the more points used in the calibration, resulting in better homographies. In this project, the chessboard has 10x7 corners.

During the calibration, the camera is static and in its setup position. It captured various frames of the plane with different positions and rotations, with the requirement of every corner being in the camera view. The more frames captured, the lower the reprojection error, and consequently, the higher the system's accuracy.

During this process, a frame is captured with the calibration plane being as close as possible to the reference plane. This plane's position relative to the camera is acquired through the calibration process and used to transform the frame pixels to the plane coordinates, using a homography. This transformation allows the computation of the pretended measurements.

4.1.2 Image Segmentation

With a high contrast between the background and the plates, it is now possible to segment the image precisely, with minimal noise. The segmentation's primary goal is to capture the plates' exact form to be processed afterwards. It converts the camera's frame to binary a binary image where the background is black, and every single pixel corresponding to the plate is white. This process needs to be consistent with high accuracy because any failure that may occur will compromise the final results due to being sensitive to errors.

From the various threshold techniques studied and tested, the one that gave the best and the more consistent results was the Li entropy-based threshold [20]. An example of this method is shown in Fig.4.1.

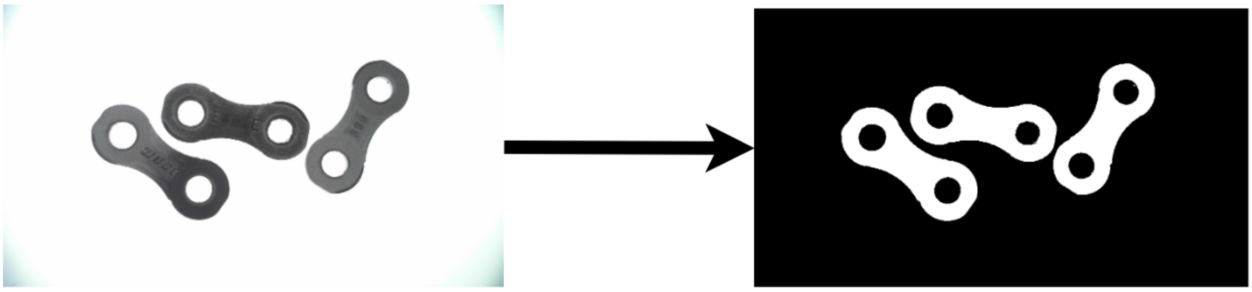


Figure 4.1: Segmentation process

4.1.3 Image-based Metrology

After acquiring the binarized image, the extraction of information is possible. Using a border following algorithm followed by a hierarchy structure [36], it is possible to know every pixel's coordinates of the plates' interior and exterior contours. The border following algorithm is applied to binary images where pixels have two intensity values, 0-pixel and 1-pixel. It searches for a connection between the 0-pixel and its neighbor 1-pixel. Once the connection is found, the 1-pixel is classified as a contour. The algorithm examines all the neighbors searching for the next connection and repeats this process, progressing through the border until it returns to the starting pixel. The hierarchy is structured by splitting the borders as parent and child. While the parent does not have any surrounding outlines, the child is surrounded by its parent's contours. This structure classifies the plates' exterior border as parents, while the plates' internal borders, which correspond to the holes, as its children. Every parent and child has a single id, so it is possible to identify various plates and analyze them individually simultaneously.

With all the borders adequately identified, the plates' measurements can be calculated. The plate's length (1.A) and width (1.B) can be measured using the Minimum-Area Encasing Rectangle [14] method applied to the plate's contours. This method consists of finding the rectangle with the minimum area that involves all the contour pixels, making the sides of this minimum area rectangle correspond to the plate's width and length. This process is also applied in the plate's children, which correspond to the holes. If the holes are a perfect circle, the minimum-area encasing result would be a square where each side would be the same size correspondent to the hole's diameter.

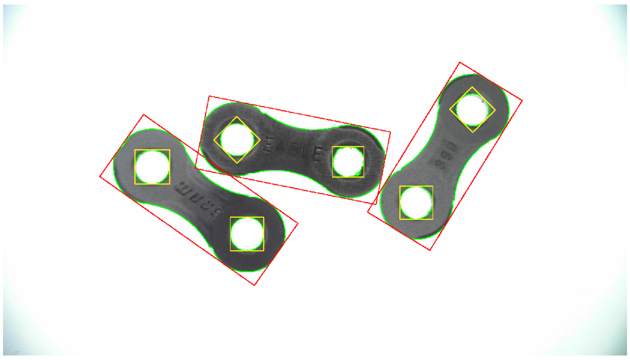


Figure 4.2: Minimum-Area Encasing Rectangle

In the next step, it calculates the centroids of every contour, using the plates' and their holes' outline, by averaging the coordinates of the borders. The centroids position (x_c, y_c) are calculated by:

$$x_c = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4.1}$$

$$y_c = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{4.2}$$

Knowing the position of the center of the two holes', obtained by the centroids, it is possible to calculate the Euclidian distance between them (1.G). With these two centroids, using the linear equation, it is possible to trace a straight line across the plate. Let $c1 = (x_{c1}, y_{c1})$ and $c2 = (x_{c2}, y_{c2})$ be the centroids:

$$y = m_c x + b_c \tag{4.3}$$

where $m_c = (y_{c2} - y_{c1})/(x_{c2} - x_{c1})$ and $b_c = y_{c2} - m_c x_{c2}$

The intersection of this line with the plate's contours results in six points, two from the parent and four from the two children. With these points, the calculation of the holes' longitudinal decentralization (1.F) is now possible. To achieve this, two of the six obtained points are used, which means that two of the four children's points need to be selected. Knowing these points relative position to the plate's exterior contour, the selection is based on the euclidean distance between them. The holes' points closer to the exterior plate's outline, are the points needed for the calculation of the longitudinal decentralization. The other two points are later used in the inscription check.

To obtain the holes' lateral decentralization (1.E), two lines perpendicular to the previous one are calculated, by intersecting with the holes' centroids. Using the equation of the previously calculated line:

$$y = mx + b_1 \tag{4.4}$$

$$y = mx + b_2 \tag{4.5}$$

Being $m = -\frac{1}{m_c}$, $b_1 = y_{c1} - m * x_{c1}$ and $b_2 = y_{c2} - m * x_{c2}$

The intersection of this line with the holes' and plates' outline, results in four points. The lateral decentralization is acquired using the same method used to calculate the longitudinal decentralization. The Euclidian distance between the points is obtained, linking the points with the smaller distance between each other. This method is applied to both holes.

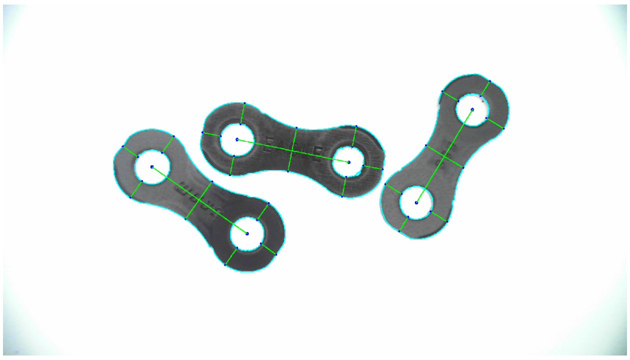


Figure 4.3: Representation of the measured lines

Every object in the captured frame is eliminated if it is not a plate, or it is a plate with significant deformations. Firstly, it analyses if the object's contour has two children, removing any that does not meet this requirement. If the object has two children, it calculates the Euclidean distance between its centroid and its the holes' centroids, comparing it with a predefined threshold. Before the system measures any of the parameters, the system converts the image coordinates to the plane coordinates through a homography, obtaining the distances in their real size.

4.1.4 Inscription Inspection

A verification of the inscription's engraving is done by reading its characters. First, an area around the inscription is selected by its relative position to the holes, using the two points acquired from the longitudinal decentralization calculation, the line (4.3) and the holes' diameter. First, two lines perpendicular to the line (4.3) are obtained (same way as lines (4.4) and (4.5)) using the two points from the longitudinal line. By using the same points and the diameter of the holes (d), a circumference equation is calculated:

$$r^2 = (x - x1)^2 + (y - y1)^2 \tag{4.6}$$

Where $r = \frac{d}{2}$ and (x1,y1) are the coordinates of the two longitudinal points. Now, intersecting the circle with the perpendicular lines, it is possible to obtain the 4 points that limit the inscription area.

$$r^2 = (x - x1)^2 + (m * x + b - y1)^2 \tag{4.7}$$

From 4.7:

$$x = -B \pm \frac{\sqrt{B^2 - 4 * A * C}}{2 * A} \tag{4.8}$$

where $A = m^2 + 1, B = m * b - m * y1 - x1, C = y1^2 - r^2 + x1^2 - 2 * b * y1 + b^2$

$$y = m * (-B \pm \frac{\sqrt{B^2 - 4 * A * C}}{2 * A}) + b \tag{4.9}$$

Resulting in the following figure.

Figure 4.4: Selection of the inscription area

Once the selected area coordinates are acquired, the area is selected in the four images with each led is individually activated. In this procedure, the color space Hue Saturation Luminance (HSL)) was used for all images. For each image, the lightness channel's (L) gradient was calculated for both axes. Using a threshold with a predetermined value, the most significant changes in intensity captured by the gradient were binarized. By applying the operator OR to all binary images, a binary image of the inscription's characters is obtained. Using the Optical Character Recognition (OCR) Tesseract, the inscription can be converted to string, verifying if it is readable.

4.1.5 Experimental Results

This system's accuracy was calculated by measuring plates with the size previously obtained using SRAMPORT's current quality control process, using the LifeCam Studio. The Image-based method developed was applied to these plates and compared to the previously measured plates, which are assumed to be the real measures. The parameters used to obtain this accuracy were the plate's height, width and distance between holes. It was observed that the system's accuracy varies depending on the plates' position. The Root Mean Square (RMS) error when the plate is centered in the capturing area is 0.0318mm. As the plate shifts from the center of the capture, the accuracy decreases, reaching an RMS error of 0.1692mm. This accuracy was computed using a plate with known sizes in 47 different positions.

Figure 4.5: **Left:** The system incorrectly measures the size of the holes and the distance between them; **Right:** System incorrectly measures every parameter.

For each capture, the system has a computation time of 225ms. The system was developed with python, mainly using the library Opencv 4.3.0, and executed on an Intel Core i7-6500U CPU @ 2.50Hz.

Inscription Inspection

The procedure developed to verify the inscriptions was tested on various plates, obtaining inconsistent results, varying from plate to plate, and proving itself unreliable as the inscription inspection method. This method is also highly dependent on the plate's position due to the LEDs pointing to the center of the capture, requiring the plates to be placed close to the center. The following results were acquired with the plate in this selected position.

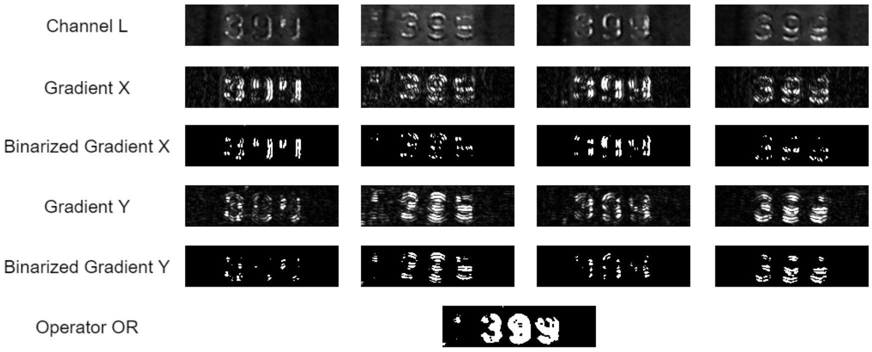


Figure 4.6: Inscription inspection procedure

The characters from the inscription in the figure 4.6 were successfully acquired, resulting in the string "399", however, this method is not consistent as a character segmentation, occurring cases where it is not able to obtain the characters in readable inscriptions, as shown in the figure 4.7

Figure 4.7: Unsuccessful segmentation of the inscription's characters

4.2 Laser Stripe Scanning

The Laser Stripe Scanning is used to verify the parameters 1.J to 1.P, completing the quality control parameters.

4.2.1 Calibration

The calibration of the LSS is similar to the image-based system, with the addition of the laser. The camera and the laser are static and in their setup position. It captures the chessboard plane with various poses, but an additional capture with the laser projected on the chessboard is done. For every plane's position, the laser plane has to pass through it. With this, the points $\mathbf{P_1}$ and $\mathbf{P_2}$ are acquired and used to obtain the laser plane equation, as shown in the section 2.4.2. Once the laser plane is known, an intersection between it and the reference plane is done, acquiring the reference plane's line correspondent coordinates. The line equation correspondent to the reference plane is obtained by applying a linear regression to the line coordinates, which is later used to calculate each point's height captured from the object's surface.

4.2.2 Plate's Reconstruction

The plate can have any rotation within the plane, but its position must be static with respect to the platform during the entire scan. In this project, the plate was positioned laterally to the laser stripe. This is due to the higher accuracy of the plane intersection with the plate than the platform movement, which depends on the marker. However, this positioning also allows the scan of the plate with fewer captures. Once all the captures required for the full scan are acquired, the segmentation is done using the method shown in section 2.4.1. The coordinates correspondent to the reference plane's projection are removed in the image frame, using the line acquired in the calibration, before converting to 3D coordinates, with the points below the line identified as noise and ignored. Every reconstruction is translated by a vector obtained between the marker position on the first frame and the respective frame, resulting in the reconstruction of the plate's surface. The laser used in this project has a wavelength of 685nm, so the image's blue channel (from the RGB format) is used to identify the marker, allowing the laser's removal in the image.

4.2.3 Experimental Results

The LSS's accuracy was acquired using a calibration piece with the thickness of 1.10mm. To do this, a reconstruction of this calibration piece was obtained using two different cameras, calculating the RMS error for each one. The cameras used were the GC650C and the LifeCam Studio, which resulted in an RMS error of 0.551mm and 0.365mm, respectively.

The LifeCam Studio had better accuracy than the GC650C due to its higher resolution, but with a field of view of 51,8º. However, the GC650C camera had a lower focal length and field of view of 33.9º, allowing a closer and more directional image of the plate. These characteristics decreased the unused pixels of capture when scanning with the cost of reducing the scanning area's size. This system has a computation time of 16s.

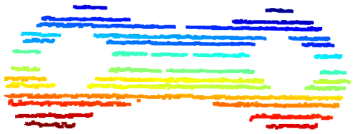


Figure 4.8: Reconstruction of the plate

4.3 Structured Light System

Like the LSS, the structured light system is used to verify the parameters 1.J to 1.P, which can complement the image-based system.

4.3.1 Calibration

In the Structured light system's calibration, a series of patterns are projected for each chessboard position, always projecting the Gray code sequence. The calibration requires the chessboard to be in the projection area for every capture, having the camera and the projector static in their setup position. The projector used in this project has a resolution of 1024x768, which led to the use of 10 vertical and horizontal projected patterns, so every pixel in the projector has a unique codeword. Considering the projector as an inverse camera, it has an intrinsic matrix and projection matrix. This intrinsic matrix calculation is done using the same method as the camera, the Zhang calibration method. When the camera captures the chessboard pattern and acquires its corners position, the corners' pixel coordinates have the codeword correspondent to the projector's frame. Zhang's method can be applied with the identified corners in the projector's frame, allowing the extraction of the projection matrices. Once the camera's and projector's projection matrices are known, a reconstruction of the reference plane is done, and its equation calculated using the PCA-RANSAC method. The equation is used to remove the plane in future reconstructions and calculate the height of the plates.

4.3.2 Plate's Reconstruction

During the scan, the plate is entirely static and within the projection area, without requiring a specific position or rotation. Once the plate is in the scanning area, Gray code patterns are projected, and the reconstruction is done. The plane's equation obtained from the system's calibration is used to remove the reference plane, resulting in the isolated plate's point cloud. Once the plane is removed, the SOR filter is applied, excluding any unwanted points.

4.3.3 Experimental Results

Like the LSS, the SLS's accuracy was acquired using a calibration piece with the thickness of 1.10mm. The reconstruction of the calibration piece was obtained using the Hitachi CP-RX70 projector and the GC650C camera. The reconstruction's RMS error was 0.365mm, resulting in a similar error as the LSS using the LifeCam Studio. The plate's point cloud acquisition can be observed in the figure 4.9.

Figure 4.9: Left: Reconstruction of the scene; **Middle:** Plane removed from the scene; **Right:** Point cloud after applying the SOR filter

The lack of points in the plane during the reconstruction is due to its low reflective properties and the camera's occlusion caused by its position. Since the projector's resolution was higher than the camera's resolution, to compensate for this difference, the camera was placed closer to the reference plane, causing an occlusion to the projection, as shown in the figure 4.10. The number of points from the reconstructed point cloud varied between 130.000 and 150.000 points in these experiments. After being processed, with the plane and noise removal, it resulted in a point cloud with 70.000 to 75.000 points. The acquisition time of this system was was limited by the projector, with a refresh rate of 80hz, requiring 500ms for the capture of all 40 patterns. The processing time for the captures was 19.8s.

Figure 4.10: Capture of a white projection, underlining the occlusion point caused by the camera

A higher intensity in the pixel projection was used to obtain the plane's reconstruction, resulting in the following point cloud.

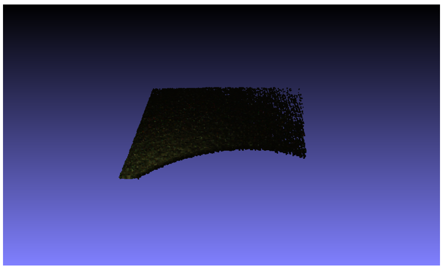


Figure 4.11: Reconstruction of the reference plane

4.4 Point Cloud Analysis

The following procedure is applied to point clouds, working for both the Structured Light System and Laser Stripe Scanning reconstructions. Due to obtaining a higher quality point cloud from the SLS, the point clouds used are reconstructions from this system. In this process, a point cloud of a model plate is used, where it is assumed to have the ideal measurements. This model plate is then compared with the scanned plate by verifying the deviation between each other's points.

4.4.1 Procedure

This method starts by comparing the model plate's and the scanned plate's eigenvalues and eigenvectors obtained using the PCA algorithm. The PCA method was chosen due to its direct and precise results without requiring any iterative calculation like Iterative Closest Point (ICP) [6] methods, which requires more processing time and is more prone to errors.

Since these vectors identify the point clouds' variances, they can be used to create an axis that consistently characterizes the plate's rotation when sorted by their eigenvalues. With this, the model plate's point cloud can be overlapped by the scanned plate. Each plate's centroids are calculated so that the necessary translation can be obtained. Then the necessary rotation is obtained using the rotation matrix of each plate. This rotation matrix is acquired using the calculated eigenvectors. With the rotation matrix and centroids of the model and scanned plate are acquired, the model plate's point cloud is rotated and translated, as shown in the figure 4.12.

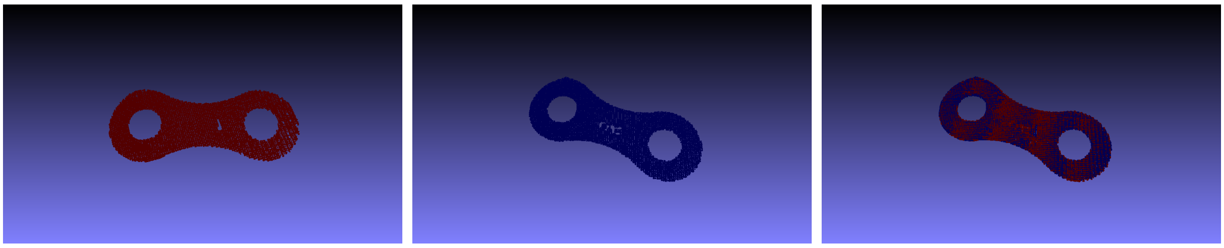


Figure 4.12: **Left:** Model plate's point cloud; **Middle:** Scanned plate's point cloud; **Right:** Both plates after applying the translation and rotation to the model

Once the point clouds are overlapped, the scanned plate points are compared with the model plate's points. Using the KD Tree algorithm, the closest point from the model plate to every scanned plate's point is acquired. This allows the calculus of the vectors that connect the two closest points between the two point clouds. The plates' difference in thickness is obtained by the dot product of these vectors with the reference plane's normal, which was previously calculated. With this, for every point in the scanned plate, a thickness deviation is attributed. To classify a point as inadequate, its thickness deviation needs to be higher than the threshold of 0.03mm, creating a selection of defective points. Some of these defective points are classified as inadequate due to noise, so another selection is made. The goal is to only select points that belong to clusters, eliminating every isolated point. This is accomplished by applying the radius outlier removal iteratively, removing isolated points. The ROR filter is repeatedly applied to the point cloud until it cannot filter more points, leaving just the clusters' points. These clusters are then used to identify the defective part of the plate, verifying the parameters 1.I to 1.P from the table 1.1.

4.4.2 Experimental Results

The system's accuracy does not permit the testing with such small deviation and does not detect defects as small as 0.03mm. Therefore, to test the detection of defects, the plate displayed in the following figure was used.

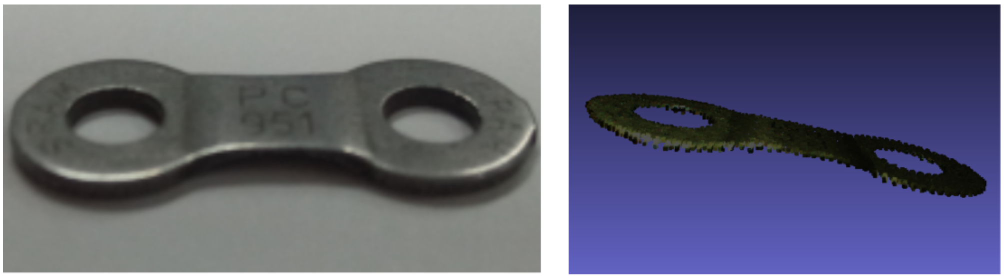


Figure 4.13: Defective board

This piece has curvature in the middle caused by a worn-out module of the stamping machine, one of the common defects in the plates. The deviation used to classify the points as defects was 0.1mm, resulting in the following inadequate points shown in the figure 4.14. This process has a computation time of 7s.

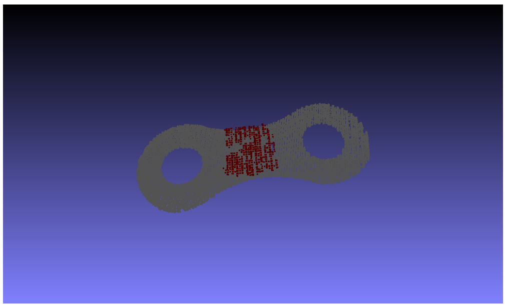


Figure 4.14: Inadequate points from the plate represented as red, compared to the model plate represented as white

5 Conclusions

This project aimed to study ways to automate the bicycle chainplates' quality control using non-intrusive visual methods. For this, three different systems were developed, analyzing different parameters required in the inspection. An imaged-based system using one camera and different angles of lightning was used to verify the parameters 1.A to 1.H from the table 1.1, while the Laser Stripe Scanning and the Structured Light System were used to verify the parameters 1.I to 1.P. The used methods can successfully identify the majority of the parameters and measure them. However, they do not have the pretended accuracy, making the system unreliable for quality control with the used equipment. The method used to verify the plate's inscription also turned to be unreliable, not having the pretended consistent results. The surface's defects detection is highly dependent on the SLS or LSS's accuracy, not being able to verify its detection in this project.

The parameters controlled by the image-based system have a maximum error of 0.015mm, while the system has an accuracy of 0.0318mm, varying with the plate position.

	System's accuracy
System	LifeCam
Image-based System	0.0318 to 0.1692mm

Table 5.1: Accuracy measuring the plates' height, width and distance between holes based on two-dimensional model

The parameters controlled by the LSS and SLS also did not reach the needed accuracy of a maximum error of 0.03mm. The LSS's accuracy using the GC650C camera was 0.551mm, while using the LifeCam Studio was 0.365mm.

The SLS's accuracy was 0.365mm using the GC650C camera and the Hitachi CP-RX70 projector. Due to the projector's focal length and camera's resolution, the capture of the full projection was not possible, with the camera only capturing a window of 131x101 pixels.

	System's accuracy	
System	GC650C	LifeCam
LSS	0.551mm	0.365mm
SLS	0.365mm	-

Table 5.2: Accuracy measuring the thickness based on three-dimensional model

All the developed systems' accuracy could be improved using a higher resolution camera, with a lower field of view and a focal length that focus on the used distances to the reference plane. All systems are highly dependent on the calibration, being limited by the chessboard printed pattern, which is dependent on the printer DPI resolution.

The main problem in the image-based system is the relation between the system's accuracy and the plates' position, where a high deviation between the plate and the center of the image is more prone to errors.

In the LSS, the main obstacle was the laser's focal length. The laser's lens would need to be adapted to work with the desired dimensions. However, the SLS suffered from a similar issue, where the main limiter was the projector's focal length, allowing to capture only a fraction of its full resolution by the camera. The SLS was more reliable from these two systems, reaching similar accuracy to the LSS using the GV650C camera. Once the plate's reconstruction was done, the developed point cloud analyzing method gave promising results to detect imperfections. This method is highly dependent on the point cloud quality of the reconstruction, being able to verify different types of plates by comparing them to their respective model plate.

For each acquisition with 1920x1080 pixels in the image-based system, there is a computation time of 225ms. While in the SLS, using captures of 640x480 pixels, the acquisition, reconstruction, and inspection of the plate took 28 seconds, compared to the operator's approximate 1 to 2 minutes.

For future work, a study about the cameras' and projector's specifications required to achieve the desired accuracy is needed before implementing both SLS and the image-based system and a new method to verify the inscriptions. These systems still need to be adapted to a continuous system, requiring the plates to always be static during the acquisition of the SLS. The developed methods can also be implemented in the quality control of other manufacturing products of different shapes and sizes, with the requirement of the product's surface to be opaque.

6 Bibliography

[1] Depth cameras for mobile ar: From iphones to wearables and beyond [online]. https://medium.com/@DAQRI/depth-cameras-for-mobile-ar-from-iphones-to-wearables-and-beyond-ea29758ec280. Accessed: October 2020.

[2] Salem Saleh Al-Amri, Namdeo V Kalyankar, et al. Image segmentation by using threshold techniques. *arXiv preprint arXiv:1005.4020*, 2010.

[3] Alex M Andrew. Multiple view geometry in computer vision, by Richard Hartley and Andrew Zisserman, cambridge university press, cambridge, 2000, xvi+ 607 pp., isbn 0–521–62304–9 (hardback,£ 60.00). *Robotica*, 19(2):88–93, 2001.

[4] Haris Balta, Jasmin Velagic, Walter Bosschaerts, Geert De Cubber, and Bruno Siciliano. Fast statistical outlier removal based method for large 3d point clouds of outdoor environments. *IFAC-PapersOnLine*, 51(22):348–353, 2018.

[5] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[6] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.

[7] Simone Carmignato, Anna Pierobon, Paola Rampazzo, Matteo Parisatto, and Enrico Savio. Ct for industrial metrology-accuracy and structural resolution of ct dimensional measurements. In *Conference on Industrial Computed Tomography (ICT)*, 2012.

[8] Fei Dai and Ming Lu. Assessing the accuracy of applying photogrammetry to take geometric measurements on building products. *Journal of construction engineering and management*, 136(2):242–250, 2010.

[9] John G Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *JOSA A*, 2(7):1160–1169, 1985.

[10] KC Fan. A non-contact automatic measurement for free-form surface profiles. *Computer integrated Manufacturing systems*, 10(4):277–285, 1997.

[11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[12] Itzhak Fogel and Dov Sagi. Gabor filters as texture discriminator. *Biological cybernetics*, 61(2):103–113, 1989.

[13] Wolfgang Förstner and Eberhard Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS intercommission conference on fast processing of photogrammetric data*, pages 281–305. Interlaken, 1987.

[14] Herbert Freeman and Ruth Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Communications of the ACM*, 18(7):409–413, 1975.

[15] Simona E Grigorescu, Nicolai Petkov, and Peter Kruizinga. Comparison of texture features based on gabor filters. *IEEE Transactions on Image processing*, 11(10):1160–1167, 2002.

[16] Banglei Guan, Yingjian Yu, Ang Su, Yang Shang, and Qifeng Yu. Self-calibration approach to stereo cameras with radial distortion based on epipolar constraint. *Applied Optics*, 58(31):8511–8521, 2019.

[17] Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988.

[18] Leiying He, Shanshan Wu, and Chuanyu Wu. Robust laser stripe extraction for three-dimensional reconstruction based on a cross-structured light sensor. *Applied Optics*, 56(4):823–832, 2017.

[19] Damir Latypov. Estimating relative lidar accuracy information from overlapping flight lines. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(4):236–245, 2002.

[20] Chun Hung Li and CK Lee. Minimum cross entropy thresholding. *Pattern recognition*, 26(4):617–625, 1993.

[21] Eugene Lin and Adam Alessio. What are the basic concepts of temporal, contrast, and spatial resolution in cardiac ct? *Journal of cardiovascular computed tomography*, 3(6):403–408, 2009.

[22] Heinz-Theo Lübbers, Laurent Medinger, Astrid Kruse, Klaus Wilhelm Grätz, and Felix Matthews. Precision and accuracy of the 3dmd photogrammetric system in craniomaxillofacial application. *Journal of Craniofacial Surgery*, 21(3):763–767, 2010.

[23] Lili Ma, YangQuan Chen, and Kevin L Moore. Rational radial distortion models with analytical undistortion formulae. *arXiv preprint cs/0307047*, 2003.

[24] Yousef B Mahdy, Khaled F Hussain, and Mostafa A Abdel-Majid. Projector calibration using passive stereo and triangulation. *International Journal of Future Computer and Communication*, 2(5):385, 2013.

[25] Suguru Miyagawa, Shin Yoshizawa, and Hideo Yokota. Trimmed Median PCA for Robust Plane Fitting. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 753–757. IEEE, 2018.

[26] DK Naidu and Robert B Fisher. A comparative analysis of algorithms for determining the peak position of a stripe to sub-pixel accuracy. In *BMVC91*, pages 217–225. Springer, 1991.

[27] Shree K Nayar, Gurunandan Krishnan, Michael D Grossberg, and Ramesh Raskar. Fast separation of direct and global components of a scene using high frequency illumination. In *ACM SIGGRAPH 2006 Papers*, pages 935–944. 2006.

[28] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[29] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, Mihai Dolha, and Michael Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927–941, 2008.

[30] Joaquim Salvi, Jordi Pages, and Joan Batlle. Pattern codification strategies in structured light systems. *Pattern recognition*, 37(4):827–849, 2004.

[31] Mehmet Sezgin and Bülent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1):146–166, 2004.

[32] Jerzy Sładek, Paweł M Błaszczyk, Magdalena Kupiec, and Robert Sitnik. The hybrid contact–optical coordinate measuring system. *Measurement*, 44(3):503–510, 2011.

[33] Irwin Sobel. An isotropic 3x3 image gradient operator. *Presentation at Stanford A.I. Project 1968*, 02 2014.

[34] Luigi Spedicato, Nicola Ivan Giannoccaro, Giulio Reina, and Mauro Bellone. Clustering and pca for reconstructing two perpendicular planes using ultrasonic sensors. *International Journal of Advanced Robotic Systems*, 10(4):210, 2013.

[35] Peter Sturm. *Pinhole Camera Model*, pages 610–613. Springer US, Boston, MA, 2014.

[36] Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1):32–46, 1985.

[37] Ming-June Tsai and Chuan-Cheng Hung. Development of a high-precision surface metrology system using structured light projection. *Measurement*, 38(3):236–247, 2005.

[38] RA Wahler and FY Shih. Image enhancement for radiographs utilizing filtering, gray scale transformation and sobel gradient operator. In *Images of the Twenty-First Century. Proceedings of the Annual International Engineering in Medicine and Biology Society,*, pages 618–619. IEEE, 1989.

[39] Feifei Wang, Lianqing Liu, Peng Yu, Zhu Liu, Haibo Yu, Yuechao Wang, and Wen Jung Li. Three-dimensional super-resolution morphology by near-field assisted white-light interferometry. *Scientific reports*, 6:24703, 2016.

[40] A Weckenmann, T Estler, G Peggs, and D McMurtry. Probing systems in dimensional metrology. *CIRP annals*, 53(2):657–684, 2004.

[41] A Weckenmann, Ph Krämer, and J Hoffmann. Manufacturing metrology–state of the art and prospects. In *Proc. ISMQC*, pages 1–8, 2007.

[42] Clarence Wust and David W Capson. Surface profile measurement using color fringe projection. *Machine vision and applications*, 4(3):193–203, 1991.

[43] James C Wyant. White light interferometry. In *Holography: A Tribute to Yuri Denisyuk and Emmett Leith*, volume 4737, pages 98–107. International Society for Optics and Photonics, 2002.

[44] Yi Xu and Daniel G Aliaga. Robust pixel classification for 3d modeling with structured light. In *Proceedings of Graphics Interface 2007*, pages 233–240, 2007.

[45] Tao Yang, Guanliang Zhang, Huanhuan Li, and Xiang Zhou. Hybrid 3d shape measurement using the mems scanning micromirror. *Micromachines*, 10(1):47, 2019.

[46] Jui-Cheng Yen, Fu-Juay Chang, and Shyang Chang. A new criterion for automatic multilevel thresholding. *IEEE Transactions on Image Processing*, 4(3):370–378, 1995.

[47] Jianxin Zhang and Alexandar Djordjevich. Study on laser stripe sensor. *Sensors and Actuators A: Physical*, 72(3):224–228, 1999.

[48] Yang Zhang, Wei Liu, Xiaodong Li, Fan Yang, Peng Gao, and Zhenyuan Jia. Accuracy improvement in laser stripe extraction for large-scale triangulation scanning measurement system. *Optical Engineering*, 54(10):105108, 2015.

[49] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22, 2000.

[50] Ling Zou and Yan Li. A method of stereo vision matching based on opencv. In *2010 International Conference on Audio, Language and Image Processing*, pages 185–190. IEEE, 2010.