

Acquisition and Reconstruction of 3D Objects for Robotic Machining

Matheus Zorawski Silva

Dissertation presented to the School of Technology and Management of Bragança to
obtain the Master Degree in Industrial Engineering.

Work supervised by:

Prof. Dr. José Luís Sousa de Magalhães Lima

Prof. Dr. Manuel Fernando Santos Silva

Prof. Dr. Marcio Rodrigues da Cunha

Bragança

2020



Acquisition and Reconstruction of 3D Objects for Robotic Machining

Matheus Zorawski Silva

Dissertation presented to the School of Technology and Management of Bragança to
obtain the Master Degree in Industrial Engineering.

Work supervised by:

Prof. Dr. José Luís Sousa de Magalhães Lima

Prof. Dr. Manuel Fernando Santos Silva

Prof. Dr. Marcio Rodrigues da Cunha

Bragança

2020

Acknowledgement

I would like to thank my supervisors Prof. José Luís S. M. Lima and Manuel F. S. Silva for their dedication, guidance, consistent support, for proposing the challenge of developing the writing of the document in English, helping with this and all other help during this work. Also thanks to Thadeu V. Brito for the help and Prof. Marcio R. Cunha for the support and incentive, been my distance co-supervisor of the Dual Diplomacy program.

Thanks to Escola Superior de Tecnologia e Gestão (ESTiG) of Instituto Politécnico de Bragança (IPB) and Industry and Innovation Laboratory (iiLab) of Institute for Systems and Computer Engineering, Technology and Science (INESC TEC) for all equipments, environment and conditions provided.

A very special thanks to my father, mother and sister, who are very important to me and provided the occurrence of this experience, besides all the support given.

Finally, thanks to all those who helped me during these 5 years of graduation, and especially to those who helped me in some way to carry out the process of Dual Diplomacy Program. Being my friends since before the university or those that the university gave me, and made these 5 years better, being in studies, laughing or conversations. Friends who, even though they were far away, were there to talk and help in difficult times. For all this my sincere thanks to Paulo, Lavezzo, Lukk, Raduy, Sergio, Maiolli, Roberta, Marina, Fábio, Will, Rafa, Rebeca, Samu, Edu, Cossich, Alberto and Horion, whom I have an enormous affection and gratitude.

Abstract

With the evolution of the techniques of acquisition of Three-Dimensional (3D) image it became possible to apply these in more and more areas, as well as to be used for research and hobbyists due to the appearance of low cost 3D scanners. Among the application of 3D acquisitions is the reconstruction of objects, which allows for example to redo or remodel an existing object that is no longer on the market. Another rise tech is industrial robot, that is highly present in the industry and can perform several tasks, even machining activities, and can be applied in more than one type of operation.

The purpose of this work is to acquire a 3D scene with low-cost scanners and use this acquisition to create the tool path for roughing a workpiece, using an industrial robot for this machining task.

For the acquisition, the Skanect software was used, which had satisfactory results for the work, and the exported file of the acquisition was worked on the MeshLab and Meshmixer software, which were used to obtain only the interest part for the milling process.

With the defined work object, it was applied in Computer Aided Manufacturing (CAM) software, Fusion 360, to generate the tool path for thinning in G-code, which was converted by the RoboDK software to robot code, and this also allowed to make simulation of the machining with the desired robot.

With the simulation taking place as expected, it was implemented in practice, performing the 3D acquisition machining, thus being able to verify the machining technique used. Furthermore, with the results of acquire, generation of toolpath and machining, was possible to validate the proposed solution and reach a conclusion of possible improvements

for this project.

Keywords: Reconstruction of objects; 3D scanner; Skanect; Intel Kinect; CAM software; Fusion 360, RoboDK; Industrial Robots in machining tasks.

Resumo

Com a evolução das técnicas de aquisição de imagem 3D tornou-se possível aplicá-las em cada vez mais áreas, bem como serem utilizadas por pesquisadores e amadores devido ao surgimento de scanners 3D de baixo custo. Entre as aplicações de aquisições 3D está a reconstrução de objetos, o que permite, por exemplo, refazer ou remodelar um objeto existente que não está mais no mercado. Outra tecnologia em ascensão é o robô industrial, que está muito presente na indústria e pode realizar diversas tarefas, até mesmo atividades de fabrico, e ser aplicado em mais de um tipo de operação.

O objetivo deste trabalho é adquirir uma cena 3D com scanners de baixo custo e utilizar esta aquisição para criar o caminho da ferramenta para o desbaste de uma peça, utilizando um robô industrial nesta tarefa de usinagem.

Para a aquisição foi utilizado o software Skanect, que obteve resultados satisfatórios para o trabalho, e o arquivo exportado da aquisição foi trabalhado nos softwares MeshLab e Meshmixer, os quais foram utilizados para obter apenas a parte de interesse para o processo de fresagem.

Com o objeto de trabalho definido, este foi aplicado em software CAM, Fusion 360, para gerar o caminho de ferramentas para o desbaste em G-code, o qual foi convertido pelo Software RoboDK para código de robô, e este também permitiu fazer simulação da maquinação com o robô pretendido.

Com a simulação ocorrendo de acordo com o esperado, esta foi implementada em prática, realizando a maquinação da aquisição 3D, assim podendo verificar a técnica de maquinação utilizada. Além disso com os resultados de aquisição, geração de toolpath e maquinação, foi possível validar a solução proposta e chegar a uma conclusão de possíveis

melhorias para este projeto.

Palavras-chave: Reconstrução de objetos 3D; Aquisição 3D; Skanect; Intel Kinect; Software CAM; Fusion 360, RoboDK; Robôs Industriais em tarefas de usinagem.

Contents

Acknowledgement	v
Abstract	vii
Resumo	ix
Acronyms	xix
1 Introduction	1
1.1 Theoretical Framework	1
1.2 Objectives	2
1.3 Structure of the Document	3
2 State of Art	5
2.1 3D Image Acquisition	5
2.2 Machining of Objects	6
2.3 Industrial Robots	6
2.4 Industrial Robots and CNC Machine in Milling Operations	8
2.5 Path Planning and Computer Aided Manufacturing	9
2.6 Related Works	10
3 Study of Tools	13
3.1 Scanning: Concepts and History	13
3.1.1 Scanning Methods	15

3.1.2	Triangulation	17
3.1.3	Structured Light	18
3.1.4	Time of Flight	19
3.1.5	Photogrammetry	20
3.2	3D Scanner Applications	21
3.2.1	Photogrammetry - Meshroom	21
3.2.2	Kinect - Skanect	22
3.2.3	RealSense - RecFusion and LIPScan	24
3.2.4	Ciclop - Horus	25
3.2.5	Tools Comparison	26
3.3	Point Cloud and STL	27
3.4	Software for Adjusts: MeshLab and Meshmixer	28
3.5	Machining	29
3.5.1	Milling Tools	30
3.6	Industrial Robots	31
3.6.1	Configuration	31
3.6.2	Workspace	32
3.6.3	Robot Path	33
3.7	CAM Software	34
3.7.1	SOLIDWORKS CAM	34
3.7.2	Autodesk Fusion 360	35
3.7.3	RobotStudio	36
3.7.4	Mastercam	37
3.7.5	Software Comparison	38
3.8	G-Code and Industrial Robot Code	39
4	System Methodology	41
4.1	Problem Description	41
4.2	Proposed Solution	42

4.3	Solutions Development	44
4.3.1	Ciclop	44
4.3.2	IRB 2600 and IRC5 controller	46
4.3.3	Tool Used	47
4.3.4	Virtual Workspace	50
4.3.5	Tool of IRB140 to IR2600	50
5	Development and Results	53
5.1	Implementation	53
5.1.1	Acquisition Steps up to .STL Format	54
5.1.2	Fusion 360	56
5.1.3	RoboDK and Real-World Adjustments	58
5.2	Tests Performed	62
5.3	Results Obtained	63
5.3.1	3D Scanner	63
5.3.2	CAM Simulation	66
5.3.3	RoboDK Simulations	69
5.3.4	Milling	69
6	Conclusion and Future Work	77
6.1	Future Works	79

List of Tables

3.1	Comparison of 3D acquisition options for this work.	27
3.2	CAM software comparison in the project criteria.	38
5.1	Estimated machining times on RoboDK.	75

List of Figures

3.1	3D reconstruction methods. Image adapted from [44]	15
3.2	Principles of a laser triangulation system. Image from [42].	18
3.3	Basic principle of structured light camera. Image from [52].	19
3.4	Basic principle of ToF. Image from [53].	20
3.5	Meshroom interface. Image from [59].	22
3.6	Microsoft Kinect sensor.	23
3.7	Small object scanner using Skanect.	24
3.8	RealSense D415 depth sensor.	25
3.9	BQ Ciclop 3D scanner.	26
3.10	Conversion of Point Cloud to STL format. Image from [78]	28
3.11	Mill tool nose types. Image from [86].	30
3.12	Typical industrial robot system.	31
3.13	Mechanical configuration illustration of an industrial robot [90].	32
3.14	Robot workspace illustration.	33
3.15	Toolpath of milling process generated by SOLIDWORKS CAM. Image from [95].	35
3.16	RoboDK library showing the variety of robots in relation to the number of axes, brand and weight.	40
4.1	Project architecture.	43
4.2	System architecture block diagram.	43
4.3	Rotating platform for stepper motor.	45

4.4	Tests with stepper motor controller.	45
4.5	ABB IRB 2600-20/1.6 industrial robot.	46
4.6	ABB IRC5 single cabinet controller [106].	47
4.7	Dremel and support for fixing the tool to the robot [8], [107].	48
4.8	Milling cutter kit and its virtual models [8], [108].	48
4.9	Creation of the milling cutter used in Fusion 360.	49
4.10	ABB IRB 2600 with the respective tool frame.	49
4.11	Virtual workspace in RoboDK.	50
4.12	Design model of the junction part.	51
4.13	Tool fixed to the IRB 2600 robot using the junction part.	51
5.1	3D acquisition process using Skanect.	54
5.2	Post-processing of .STL with MeshLab and Meshmixer.	55
5.3	Resize of the whole piece to the interest part for machining.	55
5.4	Reduction of mesh faces and converting to a boundary representation in Fusion 360.	56
5.5	Height of the workpiece and work coordinate reference point in Fusion 360.	57
5.6	First layer of a toolpath using Adaptive Clearing in in a face.	58
5.7	Toolpath of a face using Adaptive Clearing in Fusion 360.	58
5.8	Configurations of the code in RoboDK.	59
5.9	Teaching points to perform the TCP calculation.	60
5.10	Adding the real TCP and work coordinate frame.	61
5.11	Creating a file to open the program and changing the directory path to the correct one.	61
5.12	Implemented tests block diagram.	62
5.13	3D acquisition of a chair and a study table.	64
5.14	Small object 3D acquisition.	64
5.15	3D faces acquisition.	65
5.16	3D face acquisition with lots of details after mesh reduction.	66

5.17 Simulation of 3D machining strategies in Fusion 360 with a profile and diagonal view.	67
5.18 Combining surfaces and using Flow strategy with multi-axis in Fusion 360.	68
5.19 Dividing the toolpath into two operations in Fusion 360.	68
5.20 Top and bottom toolpath and respective simulation in Fusion 360.	69
5.21 Problem of generating the robot code due to the limited reach of ABB IRB 140.	70
5.22 First milling with 30 mm height and without acceleration limitation. . . .	70
5.23 Top part milling with 60 mm height and without acceleration limitation. .	71
5.24 Milling with 60 mm height and with acceleration limitation.	72
5.25 Milling with 60 mm height and with acceleration limitation of other face. .	73
5.26 Milling the top part of the new face with other milling cutter.	73
5.27 Faces in real (top images) and simulation milling(bottom images).	74

Acronyms

.STL Stereolithography file.

3D Three-Dimensional.

CAD Computer Aided Design.

CAE Computer-Aided engineering.

CAM Computer Aided Manufacturing.

CMM Coordinate Measurement Machine.

CNC Computer Numeric Control.

CT Computed Tomography.

DNC Direct Numerical Control.

DOF Degrees of Freedom.

ESTiG Escola Superior de Tecnologia e Gestão.

iiLab Industry and Innovation Laboratory.

INESC TEC Institute for Systems and Computer Engineering, Technology and Science.

IP Internet Protocol.

IPB Instituto Politécnico de Bragança.

MRI Magnetic Resonance Imaging.

NC Numerical Control.

PC Point Cloud.

PCB Printed Circuit Board.

RADAR Radio Detection And Ranging.

RGB-D Red Green Blue Depth.

SDK Software Development Kit.

SL Structured Light.

SWC Spatial Wire Cutting.

ToF Time of flight.

Chapter 1

Introduction

The intention of this work is through the integration of tools to perform reconstruction of 3D objects using industrial robots, thus dealing with 3D image acquisition, robot path planning and machining of parts using industrial robots. The basis of the work is divided into four parts, being: acquisition of the 3D image, adjustments of the acquired image, generate respective G-code, conversion to robot code, simulations of the robot code and lastly machining of the acquired part.

1.1 Theoretical Framework

The use of 3D scanners for the acquisition of objects and scenes can be carried out by different methods, which imply the accuracy of the acquisition, time of acquisition and processing, price of the equipment and the software available for that method or device. Due to the fact that 3D scanners are used in several areas, they have evolved considerably to be able to cover other applications and be used with a new purpose, such as the emergence of low-cost versions that can be used by hobbyists and researchers.

In the manufacturing process to generate machining projects with a higher level of complexity manually, it becomes very difficult, however, with the use of CAM software the complexity can be greatly reduced, and all the necessary path for machining of Computer Aided Design (CAD) model file can be generated. However, most of these software

programs are aimed at Computer Numeric Control (CNC) machines, which are used in machining services and have a high level of rigidity, that allows machining different types of materials. However, CNC machines are only intended for machining operations, have a high initial cost and limited working area.

Industrial robots are highly integrated into various sectors of the industry, having a low initial cost in relation to its functionality, with a wide working area and multi-axis of work, even with low rigidity compared to CNC machines, industrial robots can be used to machine parts, just requiring a tool for machining and the robot toolpath for that.

1.2 Objectives

This project has as main objectives to perform machining of parts acquired by a low-cost 3D scanner, using an industrial robot, thus performing 3D parts reconstruction. Therefore, sought to achieve the following steps:

- Study of 3D acquisition methods.
- Perform 3D scanner tests.
- Choose a low-cost 3D scanner to use in the next steps.
- Finding a way to adjust the acquired 3D image.
- Study of CAM software programs for machining operations.
- CAM software programs selection and tests.
- Generate G-code of the acquired part.
- Convert G-code to robot code.
- Perform machining tests of the generated code.
- Machining the part in a real environment, using an industrial robot.

1.3 Structure of the Document

This document is divided into 6 chapters, to describe the work developed throughout the dissertation.

Chapter 1 contains the general introduction of the project, thus dealing with what is intended to be accomplished in this work, a brief explanation of the reason for the work and the objectives to reach the solution of the problem.

Then in Chapter 2 the state of the art is presented, with the necessary bibliography for the development of this project, showing the history of the evolution of the tools to make possible to perform this work, together with the motivation and related works.

Chapter 3 presents the tools and theory necessary to carry out this work, covering from 3D acquisition methods to machining converted into robot code and knowledge of the tools for this process.

The description of the problem is presented in Chapter 4, where it deals with the system methodology, presenting the proposed solution to the problem and the necessary development of the tools studied in the previous chapter to achieve this solution

The steps for implementing this work, together with the tests carried out in all stages and the results obtained from these tests are presented in Chapter 5.

Finally, Chapter 6 presents the conclusions of this work and ideas and suggestions for future work.

Chapter 2

State of Art

This chapter will address the State of the Art of this research, that is, the bibliographic references for the development, choice of tools and the scope of work. For this, it is divided into six sections, the first one (2.1) presents the history and techniques of 3D acquisition, then in Section 2.2 the importance to produce objects is introduced, followed by the history of the evolution of industrial robots, with their applications and advantages in section 2.3. In the next section (2.4) is presented the concept of CNC and compared to industrial robots in machining application, than deals with the concept and the evolution of the path of industrial robots in Section 2.5 and finally, the related works are introduced in Section 2.6.

2.1 3D Image Acquisition

In the early 1950s, research laboratories began to use computers for the processing of several data, and by the mid-50s, the possibility of applying computers for image processing was noted. However, for computational 3D reconstruction, huge computational processing is required, for at least a reasonable image processing, which was only possible in the late 1950s, due to the evolution of computers in storage and memory [1].

The techniques that are used to acquire a 3D image can be based on physical contact (which can be destructible or not) and without physical contact. Among the non-contact

techniques, the ones that stand out the most in the various applications are those by optical ways [2], due, in addition to not needing physical contact and being a non-destructive feature, also of their purposes of applications and acquisition way to the signals to be processed [3]. Optical scanners include triangulation, time of flight, structured light, interferometry, stereo, movement, silhouette, texture and shading methods [4], [5].

Nowadays, due to the different technologies that have been used to carry out the image 3D acquisition, and the different existing methods of acquisitions, there was the development of scanning devices. The vast majority of 3D scanners already contain good accuracy, speed and efficiency to create 3D models, which can range from bench scanners with good accuracy, scanners to operate over long distances and low-cost scanners. These facts make it possible the apply 3D scanner in new areas [6].

2.2 Machining of Objects

The need to mechanize the production of objects has existed since the beginning of humanity, in which initially were used fully manual tools to carry out this production. With the need to increase the scale and efficiency of the piece producing, machine tools emerged, which made it possible to carry out these processes with greater mechanical assistance and without a total human intervention needed [7].

In the future, with the technological evolution of the '60s, when Direct Numerical Control (DNC) appeared and of the '70s, with the emergence of CNC, there was a replacement of conventional machine tools by machining centres. These machining centres are modern machines that do completely automatically the machining of pieces, hence being independent of the operator during the manufacturing process [7], [8].

2.3 Industrial Robots

The first appearance of industrial robots had low technology, and can be said that it originates from the junction of the hydraulic assembly machines and the Numerical Control

(NC) machines, thus being initially hydraulic and shortly thereafter also pneumatic, did not contain communication with the external environment and carried out really simple actions, such as loading, unloading and materials handling [9]. The “Programmable Article Transfer” projected in 1954 by George Devol, allowed in 1961 the creation of Unimate, which is considered the “true” first industrial robot, where in the same year of its appearance, it was installed in an automobile company, and soon spread to other automotive factories [10].

A major change in robotic manipulators occurred in the late 1960s, where, unlike the first manipulators that needed the presence of a human to perform an action regularly, their specific path started to be programmed which could be repeated by performing a certain task forever, without the need of a human to perform the action. In addition, another major breakthrough for manipulators was the introduction of microprocessor in 1976 by Intel, which made it possible to record the path of the manipulator in the desired task, thus allowing users to teach trajectories to manipulators [11].

It is no longer possible to imagine the industrial company today without the existence of industrial robots [12], where these go beyond just reproducing pre-programmed tasks [11], being used in areas such as painting, welding, machining, coating and sealing, and assembly. Because robots have high speed, precision and accuracy characteristics, one of their most frequent applications is welding (spot, arc or laser) and spray painting [12].

In studies like of Santos, indicate that because the industry seeks to produce in high quality, quantity and with low cost, the use of industrial robots in machining has become viable, as they contain a wide working area and great flexibility [7]. Also Angelino shows the emergence of studies on the use of industrial robots in finishing operations (removing burrs, sanding and polishing parts), to take advantage of its high level of accuracy and precision [13].

2.4 Industrial Robots and CNC Machine in Milling Operations

In the 1950s, there was the emergence of machines aimed at automating finishing operations, but without success due to the high cost to change the desired operation. Only later, with the emergence of the CNC machine turned possible to achieve good results in the production and finishing of pieces automatically, as these machines have a high level of rigidity allowing to make great contact forces, in addition to easily changing the desired operation. But due to their degrees of freedom and the maximum number of operations performed (usually one), these machine have limitations of flexibility and machined geometries, hence keeping the need for human labour in some of their applications, in addition to requiring high investment and carrying out only a group of limited actions [14], [15].

Robotic arms are composed of rotation and translation joints, but even so, their load capacity, rigidity and precision in operations are lower compared to other multi-axis machines (such as NC). This happens due to the nature of their joints being sequential [11], therefore, it is estimated that a CNC machine is approximately 50 times more rigid than a robotic arm, which makes robots application difficult to machine very complex or rigid material and surfaces [13].

The industrial robots characteristics of highly flexible and a low initial cost, compared to CNC machines, apart from the requirement for production lines to have less and less dependence on manual operations, are stimulate points for studies using industrial robot applications in new areas [13]. According to Barbosa industrial robots tested in companies for finishing operations point out to have a greater advantage than disadvantage. Same author presents a study about industrial robots for sanding processes of components, which is one of the possible finishing operations [16].

Study on rigidity of industrial robots from Mejri *et al.*, indicates a difference in stability depending on the radial direction of the operation and the position of the robot, thus requiring adjustments of the cutting conditions according to the change in the robot's

posture [17]. Robot manufacturers are proposing optimizations in the design of the robotic structure, joints and power transmission, for better applications of robots in machining tasks. Besides that, there are recent researches for better stability, such as vibration predictions during machining [18], tests using robots with greater rigidity such as IRB6660 which has a structure aimed at greater stability [19] and proposals to compensate and reduce precision errors due to the force exerted on the tasks [20].

Another major barrier to the use of industrial robots in machining or related cases, is the translation of CAD models into a robotic trajectory. However, with the evolution of CAM software, industrial robots have been applied in several sectors, like in finishing tasks, as previously mentioned by [16] (sanding) and [13] (deburring), in addition to other areas, such as machining to make custom pieces [21], milling by converting the generated G-code [22], Spatial Wire Cutting (SWC) using two 6-axis robotic arms to shape a material with the curvature of a hot wire [23]. Industrial robots have been also applied in additive manufacturing from products with complex geometries that need to be post-processed by milling techniques [24], to produce printed pieces larger than conventional ones by 3D printer [22] and printing in different directions by decomposing the model to be printed without needing a support structure, increasing the flexibility of 3D printing [25].

2.5 Path Planning and Computer Aided Manufacturing

For a robot to perform a task, it must move through points in space, along a defined trajectory, with determined speed and acceleration, with certain interactions between the robot and the points visited [13]. The main method of programme robots are manual, moving the robot in determined positions and saving these coordinates, called this method by “teach and repeat”. The accuracy of this method is generally low, the opposite of CNC machines, which are programmed offline and based on known reference coordinates [26].

A programming method for industrial robots was proposed in 1999 by Hu and Chen using the x , y and z directions, representing the tool and the model of the part to be cut in a grid array [27]. In the same year Hu and Chen changed the grid resolution to obtain more accuracy, but the grid size affect the real size of the program, which in some cases it is not possible to be handle by some controllers [28]. Also, it was proposed in 2004 a setup using dual robot, with one robot to handling the material and the other one to bearing the tool [29].

Laboratories, industries and other environments in which machining tools operate need computational assistance. It is where CAM has become increasingly popular over the years. These software aim to reduce the work of the manufacturing engineer and the operator, in order to reduce the cost and time of projects, aiming to guarantee a collision-free interaction between the tool and the workpiece or the path executed by this tool, all generated as automatically and quickly as possible, and without the need for the operator to have a depth knowledge programming [13], [30].

Publish about automatic robot machining path planning is very lower compared with CNC machining path planning. There might be a misconception thinking that CNC path planning are the same or similar to robot machining path planning, but the difference is substantial due to the impact of stiffness on it [31]. Some companies and academic research are engaged in developing software packages capable of generating robot trajectory automatically from a CAD model since 1994 [32], the most part of articles for planning in robot machining are using methods based on Cutter Location by an existing method, or an existing CAD/CAM software package considering the robot dynamics [31].

2.6 Related Works

3D acquisition technologies are becoming more and more accessible to the hobbyist, researchers and public, as Moreira seeks to use free software for low-cost 3D acquisition through the triangulation method. Oliveira uses an embedded system to automate the

acquisition of images for 3D acquisition using photogrammetry, and the techniques continue to be improved, as Liu and Lin test high-order sinusoidal patterns in addition to conventional sequential binary patterns to improve scanning and reduce errors in the 3D acquisition by Structured- light [3], [5]. In addition to these projects, there are projects dealing with the integration of 3D acquisition with a 3D printer, such as Duarte [34], or in CNC machining, that with a CAD model properly acquired and treated, Santos uses a CAM software to obtain the G-code, and then machining the part on a CNC machine [7].

Besides to the existing CAM software to generate the toolpath in conventional G-code of a 3D model, a research was already carried out in 2009 by Miranda to generate it in an automated way [35]. Also, the CAM software have been used with industrial robots generating robot code for finishing operations, such as by Angelino [13] using a robot for guitar finishes, or in 2014, when Malheiro worked on the optimization of finishing process with robots through a force control system [14].

CNC machines and 3D printers have range limits, so some proposals appear to try to improve these technologies by testing new materials and finding ways to perform these jobs. A method that has been researched to increase flexibility and widen the work area, which consequently increases the maximum possible machining range with these technologies, is the inclusion of industrial robots in the execution of these tasks. As in 2013, when Oliveira used an ABB IRB 2400/16 robotic arm to do machining on 3- and 5-axis of complex surfaces, obtained in CAD formats and, in 2018, Pollák *et al.* used an ABB IRB 140 robotic arm and adapting a structure of print head to perform 3D printing at scales larger than a conventional 3D printer [36], [37]. Also, industrial robots were used to perform 3D parts machining from a CAD model by a Nachi SRA166 robot arm in 2020 by Prasai *et al.*, and Miranda in the same year used an industrial robot to manufacture of customised items using photographic record to acquire formats of the items [21], [38].

Chapter 3

Study of Tools

Chapter 3 presents the theoretical and technical knowledge necessary to understand and develop this work. Initially addressing the study of scanning and its methods in Section 3.1, followed by the applications of scanning standard methods in Section 3.2. Concluding the part of the 3D acquisition process in the following sections, where in Section 3.3 deals with the output generated from the scanners and in the next Section, 3.4, the software to adjust these outputs is addressed. Finally, the concepts of machining process and industrial robots are covered in Sections 3.5 and 3.6 respectively. The related software to assist this process is presented in Section 3.7 and Section 3.8 stresses at first the creation of the toolpath from a CAD model and examples of these software, and at the second, the conversion into robot code. It is also addressed the difference between these technologies and software, and the considerations for the choices between them in the developed work.

3.1 Scanning: Concepts and History

Among the various applications of reverse engineering to reconstruction and conversion of physical objects to a virtual environment, day after day it is gaining importance in various fields, such as entertainment and everyday life, in research and especially for the various sectors of the industry, as with the virtual version of a piece it is possible to reproduce, modified or refined it [39].

To analyze data on a simple geometry object it is possible to perform measurements from basic tools and then create the digital model. However, when objects have more complex shapes, this method by measurements becomes unfeasible and requires a long time to obtain the digital model, where the need for scanning objects using surface mapping methods comes in [40].

Even though the search for distance forms and standardise measurements has occurred since before 240 BCE, in which the first methods required direct contact with the surface to be measured. In 1603 Christoph Scheiner invented the pantograph, which used a mechanism to accurately replicate an existing design, and has the same principle as the Coordinate Measurement Machine (CMM) invented later. During the industrial revolution (1750 - 1850) to make copies of precise parts, advanced measurement methods needed to emerge to guarantee mass production. The contact-based methods can cause damage to objects and require a lot of time to obtain an accurate 3D model, so non-contact scanners emerged to work with these problems [41].

The non-contact and contact methods use the same idea for the reconstruction of a 3D model, where they use a reference point to calculate where each point is on the surface of a given object in relation to the scan part, thus generating a cloud of points with the shape of the object, which can be processed using software to perform a virtual reconstruction of the object [42]. So a 3D scanner is a device which collects silhouette and shape, sometimes together with colour, from a certain environment or real-world objects, having its certain advantages or disadvantages according to the acquisition method used [33], [43].

Some of the important parameters for a scanner are the scanning speed, accuracy and repeatability, resistance to environmental conditions and the maximum scan volume. All of these parameters depend on the methods and technique used to acquire the shape of an object, some being very precise, enabling accuracy to be less than 0.01 mm, and others making it possible to acquire large objects [39].

3.1.1 Scanning Methods

There are several ways to divide the 3D scanner methods, such as manual and automatic, and according to Cheng are first classified into contact or non-contact methods [4]. Besides that Sousa Azevedo, Curless and Seitz [1], [44] divided the definitions of three-dimensional acquisition between the different techniques as the contact methods: CMM, Jointed arms and Slicing; and the non-contact ones: Stereo, Shape from shading, Shape from motion, Shape from silhouettes, Depth from focus/defocus, Imaging radar, Triangulation, Interferometry (Moire and Holography), Active variants of passive methods (such as Active depth from defocus and Active stereo), Time of flight (ToF), Industrial Computed Tomography (CT), Ultrasound, Magnetic Resonance Imaging (MRI), as depicted in Figure 3.1.

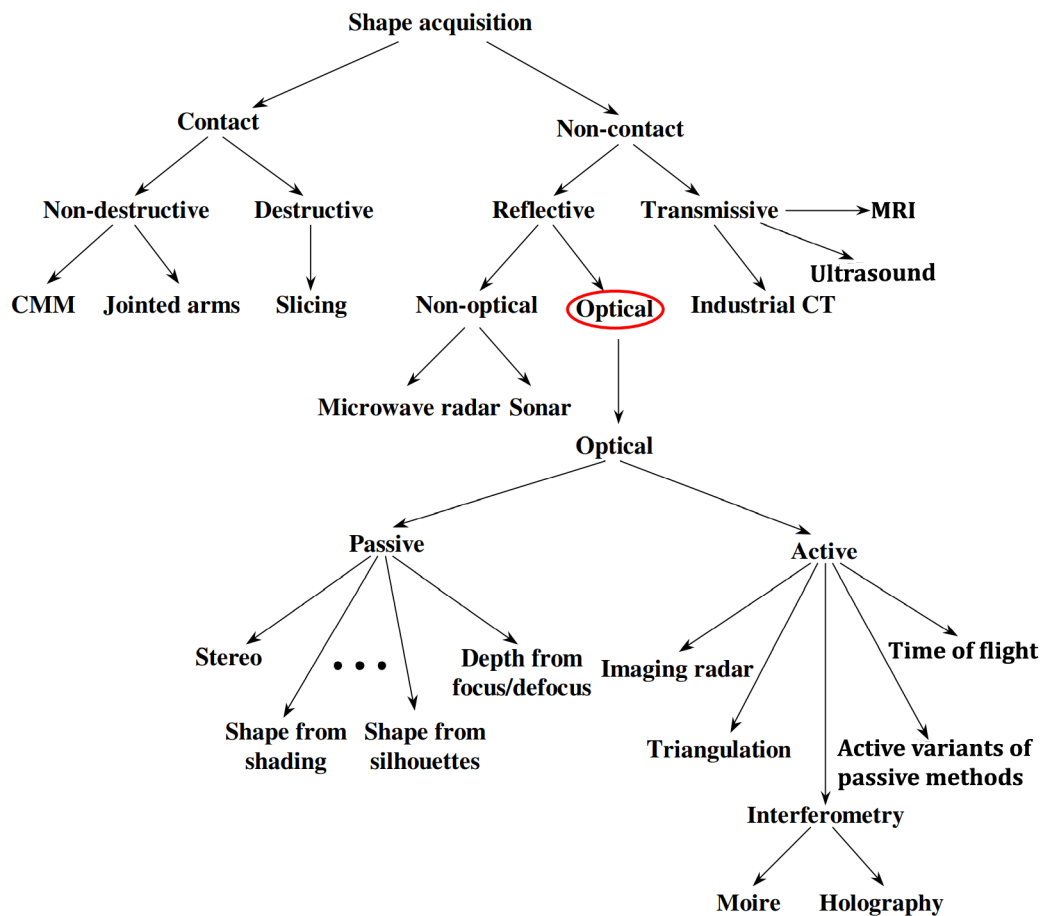


Figure 3.1: 3D reconstruction methods. Image adapted from [44]

The contact-based methods, as its name implies, is when it is necessary physical contact with the object that seeks to be reconstructed, that is, a method that collects data set by mechanically moving the scanner's reading point, to acquire the relevant regions to be measured. These methods have a wide variety of applications and can achieve a high accuracy rate [1]. However non-contact methods use image acquisition, emitters and detectors as sensors to be able to acquire data without the need to touch the object [1].

Usually non-contact methods are divided between active and passive, where the first, needs to emit some type of energy (structured light, x-ray or ultrasound) or a movement between the object and the acquisition system, which can be brief by some authors by the need to control ambient light using a illumination source, avoiding some problems of the only passive stereoscopic system [1], [41]. Although these methods are generally more accurate than using just passive techniques, and can facilitate the computation of the scene, these techniques, such as laser scanning, tend to be more expensive, more delayed, and more intrusive than their passive equivalent methods, beyond this are not always viable, in cases of distant or fast-moving objects modelling [45].

Passive methods do not require the emission of energy or generation of movement, basically just use the information and light available in the environment, what makes these methods cheap, due to needing only a camera to capture the images necessary for the reconstruction of objects [45], [46]. On the other hand, these methods, in the case of triangulation or others that use cameras, need to find correlations between the various points acquired, thus requiring a high level of processing due to the computer vision being robust. But there are ways to control some conditions, such as using a known or constant background, that helps to identifying the external boundaries of foreground objects in a reliable way [41].

Normally some of non-contact methods that require image data, need relevant resources to the correct input, which in cases where are occlusion cases, lack of significant features on the object's surfaces or low texture, can be difficult or impossible to acquire the correct data [1], [47].

Due to the advantages mentioned about non-contact methods, they are the most used and researched, and among them, the most prominent of used are ToF, Photogrammetry, Structured Light (SL) and Triangulation, which will be best explained below [3], [33].

3.1.2 Triangulation

Triangulation is the most common short-range scanners method, which uses trigonometry to calculate the position of 3D points, that are illuminated by a pattern of light on the object surface, that desire to make a digital version of this object, that is, make a 3D scan. Normally the light pattern is a line or spot produced by a laser, or a pattern form from a light source that uses a mask [42].

In this method, as already told, a laser beam is projected over the object, and a sensor (normally a camera), captures the projection of the laser into the object, and then the image will be treated by the software, that searches in this image for parts with the colour patterns of the laser, so the treated image will result in the outline of the object in that position, in relation to the laser, as can be visualized in Figure 3.2. Thus, this process is repeated by rotating the platform, where the object is located, and saving the specific contour for each position. This method may need more than one scan according to the complexity of the object, to acquire all the desired faces, which must be defined by the user [34].

After the acquisition of the contour points, it is necessary to convert them into three dimensions points, of a matrix with depth values. For this, the distance of each contour pixel must be calculated, the sensor is located at a known distance from the laser source and the angle of rotation of the laser is also known, with trigonometry it is possible to calculate the real coordinate points, and thus acquiring accurate data from the various points on the surface of the object [48].

With the depth matrix values, some software is used to map the points acquired by the various acquisitions of the object into a point cloud, which represents the 3D object. So with this Point Cloud (PC) it is possible to use algorithms to connect all of these

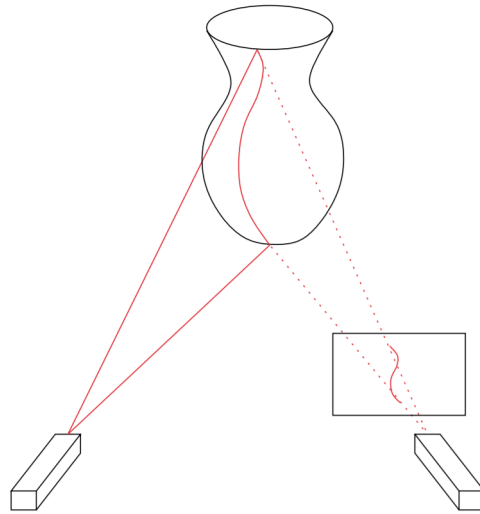


Figure 3.2: Principles of a laser triangulation system. Image from [42].

points and produce a polygonal mesh structure, that can be saved as a Stereolithography file (.STL) [34].

The properties of the surface to scan affect the accuracy of results and triangulation scanners can have quite problems with materials that have shiny, high light absorption or transparent surfaces. In addition, this method works in a short-range, so it is possible to scan only small pieces. However, this method has a good resolution, accuracy and reasonable price, due to its design can be quite simple [42], [49].

3.1.3 Structured Light

SL also uses trigonometry on its method, but different than a normal triangulation method, which uses a projection of a line or dot onto the object, this method works by projecting a coded light pattern. The advantage of this method is the possibility of capturing the entire surface of an object and not just a line, increasing the capture speed, thus being able to deal not only with static acquisitions but also, depending on the technique used, allow 3D reconstruction in real-time for a moving targets [50], [51].

The first methods of structured light coded appeared in the 80's, being improved later on in the 90's, focused on reducing computation time and increasing depth resolution [2],

[5]. These techniques are classified into single-shot and sequential methods, using a single pattern or a sequence of patterns onto the object to be mapped. The second method is more accurate, but takes more time to 3D reconstruction, on the other hand, the single-shot method enables real-time 3D reconstruction [2], [50], [51].

One or more sensors/cameras are placed away from a projector, and as in the triangulation method, distances and angles between them are known, so that it is possible to calculate the distances from the light pattern points that are in the field of view, as the configuration to exemplify this method on Figure 3.3. The projected lights can have different colour and shape patterns, which depends on the method and technique used, however, these lights use normally patterns encoded by strip or grid indexing [5], [51].

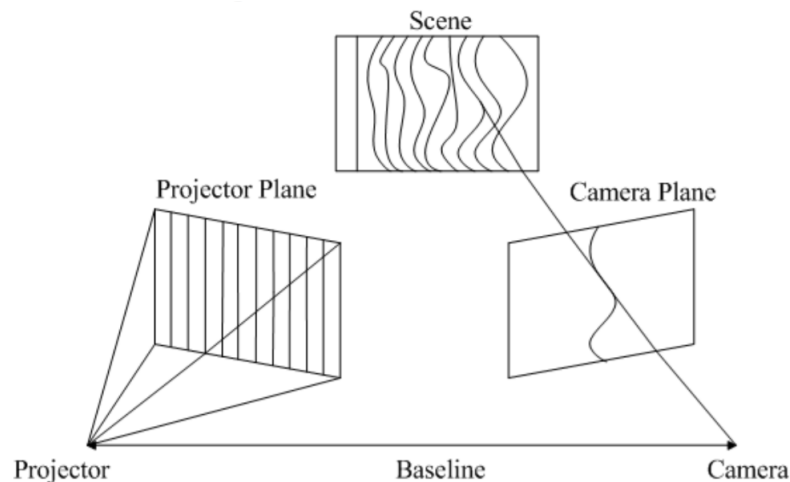


Figure 3.3: Basic principle of structured light camera. Image from [52].

3.1.4 Time of Flight

Until not too long time ago the technology and instrumentation used to do dynamic scene acquisition were only accessible by research labs or big companies, which with the insertion of Time-of-Flight range cameras in the market recently, become it widely available to use depth data in generic or dynamic scenes [53].

The method Time-of-Flight (ToF), as belonging to the optical methods, are non-contact, also reflective and active, as seen previously on Figure 3.1. The difference between

the last two methods treated is that it does not use the capture of images of objects with patterns on them, but the method uses the Radio Detection And Ranging (RADAR) principle, where emits electro-magnetic radiation (that travels at a known speed in the air, the light speed, approximately 3×10^8 [m/s]), and with this information, calculates the time to this radiation achieve the scene, it is then reflected back by the scene surface, and travels back to ToF sensor receptor, which is normally positioned with the transmitter, as the Figure 3.4 [53].

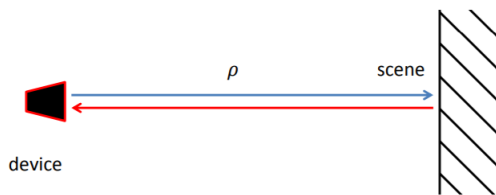


Figure 3.4: Basic principle of ToF. Image from [53].

The acquisition of the geometries can occur using point-wise ToF sensor systems together with a mechanism to perform the scan in time-sequential or with the matricial ToF cameras, which are some rows and columns of independent sensors, which makes it possible to read distances from a scene simultaneously, and provide these distances by correlating each pixel in a matrix with a corresponding point in the scene. In the first case, as being a time-sequential scanning, it is not good to acquire dynamic scenes, because it takes some time to scan a scene, that is, unsuited to acquire moving objects [53].

3.1.5 Photogrammetry

A way of three-dimensional reconstructing of objects or scenes is through photographs of digital images, without using an active method as mentioned above, but rather a passive reflective method, that is, it captures scenes using reflected environment visible light. [54].

The principle of photogrammetry is to use several photographs of a static scene, taken from different viewpoints, and with computer vision and computational geometry algorithms, detect automatically pixels corresponding to a same physical point of the scene in different views. For that, it is required user input data with the parameters of the

camera, such as focal length, lens distortion and other items essential for the calibration, which allows to control the accuracy of the acquisition by changing this parameters and the resolution of the camera [55].

A great advantage and difference of using photogrammetry technology is the capability of the various scale of reconstruction, with a good precision. This method provides a permanent record, which makes possible to add new data for a specific desired part, besides being a non-contact method that offers real-world colour and texture. For the other hand, one downside of the method is the time taken to run the algorithms to reconstruct the scene, there are always oclusions or obstructions on the images and the accuracy can degrade substantially if there is any change in the scene during the process of image acquisition [55], [56].

3.2 3D Scanner Applications

For all the scanning methods mentioned in the previous chapter, there are different forms of applications, where for each method and technique of this method, there is a specific hardware and software solution, which all seek in some way to measure the depth of the various points desired to be acquired, so that with these points, it is possible to determine the shape of the location or object [5]. Some hardware and software applications were studied for the execution of the work, turning to the available and low cost tools, which will be presented below.

3.2.1 Photogrammetry - Meshroom

Meshroom is an open-source software for 3D Reconstruction using the photogrammetry method, based on the AliceVision framework and according to Karnicki due to its features is a great solution for a low-cost 3D scanner, in addition to being solution with good results and a completely open-source. The system to reconstruct images on 3D objects uses an algorithm to compile by a simple node-based methodology which connects its necessary steps. The interface is friendly as can be seen in the Figure 3.5, where shows the pictures

added, a selected picture and the point cloud generate with each point of view respectively, and it is possible to add more pictures to improve more detail in some area during the half-finished solution [57], [58].

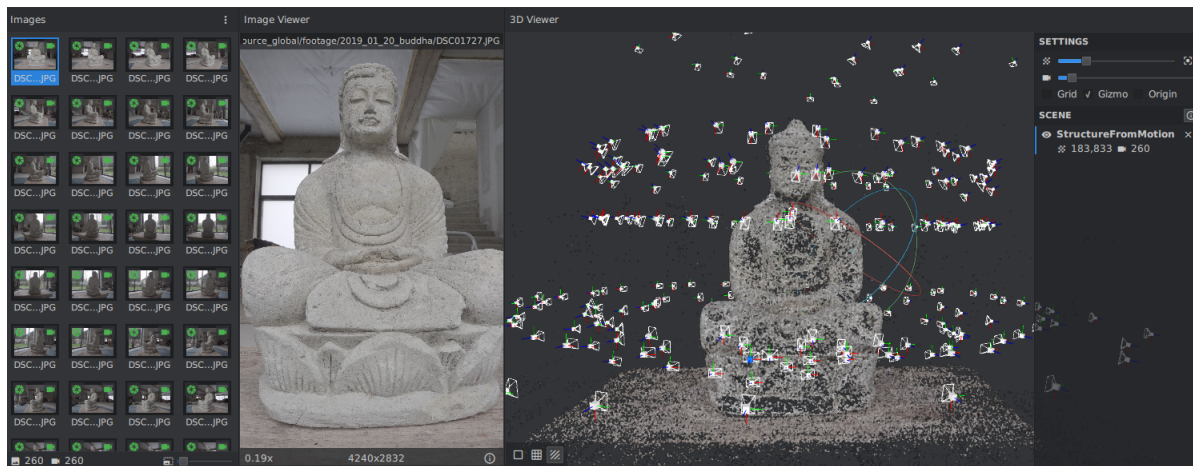


Figure 3.5: Meshroom interface. Image from [59].

The difference between Meshroom and other programs available for photogrammetry is the simplicity of operation, where just need to add pictures into the Meshroom, hit start and wait for the finished 3D model, which, in comparison with other programs, must operate each step individually. Meshroom also shows all the steps forming a pipeline of nodes, and the user has the option to adjust as desired to get a specific result tweaking the parameters from its created pipeline, such as edit the texture step to set the resolution of the texture map [58].

The program allows for obtaining textured 3D models and the point clouds are stored in the Alembic file format (*.abc), which can be post-processed in MeshLab or CloudCompare. A downside of the program is the storage, which requires a lot of space, as a project of with about 300 photos can take more than 50 GB [39].

3.2.2 Kinect - Skanect

The Microsoft Kinect was developed with the main function of motion detection, announced in 2010, when started a revolution in games in the part of interaction, due to

its sensors that captures movements and voice commands of the user. After the company saw the potential of Kinect for other applications outside of the games area, due to some researchers using this technology, the company made the Software Development Kit (SDK) to be used by developers in their own application [60], [61].

The release of Kinect and other Red Green Blue Depth (RGB-D) cameras has made the 3D reconstruction possible for a low price, and even real-time 3D reconstruction with the aid of some algorithms, where the device can be connected to a computer easily using the USB interface, and then use software to take advantage of its features through algorithms [62].

As shown in the Figure 3.6, the device has an infrared emitter, an RGB camera, an infrared depth sensor and arrayed microphones and accelerometers, besides a motor to adjust the angle of view vertically [63].

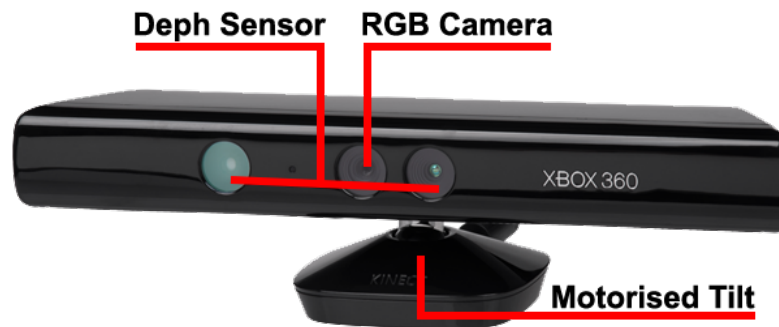


Figure 3.6: Microsoft Kinect sensor.

The RGB camera captures colour images with a resolution of 1280x960 or streaming images at 30 frames per second with a resolution of 640x480. The depth sensor uses the projected pattern to extract the depth information from the scene, in which the depth camera has a resolution of 640x480 [63]. These features allow Kinect to be used to capture geometric information from the object's surface through depth images, fulfilling the function of a 3D scanner and, at the same time, reaching high-resolution of colours by the digital camera [64].

Skaneet is a software that uses Microsoft Kinect or Asus Xtion to capture the 3D model (with or without colour) of an object, which makes it possible to create 3D polygonal

meshes of real-world objects in few minutes [65]. It is developed by the North American startup Occipital, which makes it available for a value of 119€, but it also has a free version that allows to work with some restrictions on the maximum number of polygons for export [66].

The program works by creating the digitalized surface as data is obtained with the Kinect camera, which is recorded to be better processed after the acquisition is completed. For the acquisition, the object of interest must be contained within a pre-defined area in the program, in addition to not making sudden movements in the object and Kinect in this process, a way to facilitate acquisition is to rotate the object of interest on an axis and keep the device fixed [66]. Figure 3.7 is an example of acquiring Skanect of a small object.

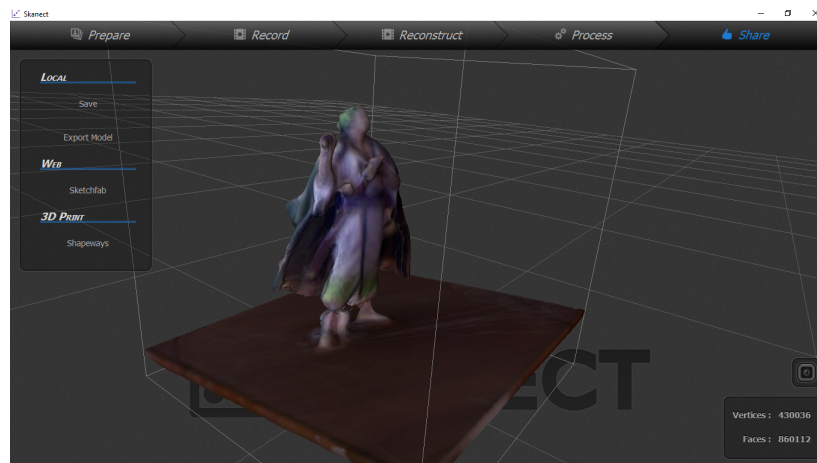


Figure 3.7: Small object scanner using Skanect.

3.2.3 RealSense - RecFusion and LIPScan

Intel RealSense D400 family offers two versions with depth camera: D415 (Figure 3.8) and D435 that brings the most advanced depth perception capability, integrating a processor to handle the complex depth algorithm, in the optical part has 2 image sensors of 1080p, with infrared projector and a RGB image sensor to colour data, acquiring data placed between 50 cm and two meters away from the camera with an accuracy of 2.5 mm to 5

mm [67].



Figure 3.8: RealSense D415 depth sensor.

The D400 series uses a hybrid method as it uses the ambient light and the projected pattern when necessary. It also uses SL and ToF, by collecting light emitted in the depth camera with the respective time of reflection (ToF, or using the image and the triangulation method to calculate the structured light. To compute most of the depth metrics and indicate the basic depth image quality of any RealSense D400 family, the Intel RealSense SDK 2.0 is used, that is an open source software provided by Intel [67].

To use RealSense for 3D reconstruction some software options are RecFusion and LIPScan. According to Omori the first software is used for having a clean interface and good results compared to similar software, but both can export the 3D model in formats .PLY, .OBJ or .STL files. The main difference between them is that RecFusion works only needing a depth-sensor like the Microsoft Kinect or the Asus Xtion, and the LIPScan is fine-tuned to complement Intel RealSense D415 Technology, aiming for better measuring, or digitizing processes. Both are good for making the acquisition on a rotating platform, LIPScan has a limited area acquisition smaller than RecFusion, but has free exports, unlike RecFusion [69], [70].

3.2.4 Ciclop - Horus

Ciclop is a 3D scanner based in triangulation, in which its design of the mechanical and electronic part is totally free and also the software to perform the acquisition, which allows sharing and reworking, just it requiring that derivative works be licensed under the same license, even for commercial use [5].

The device structure has two laser emitters, a rotating base, a stepper motor, a webcam, a BQ ZUM CORE (microcontroller derived from Arduino UNO) and BQ ZUM SCAN (a shield), as shown in Figure 3.6. Its control software is based on Arduino, due to its microcontroller, and ZUM SCAN is a CNC shield. The kit with all mentioned parts without the structure and rotating base, has a cost of 159.90€, being marketed exclusively in Europe [5].



Figure 3.9: BQ Ciclop 3D scanner.

Horus software is available for Windows, Linux and MacOS operating systems, being open source and having its main code available for download, which was developed by Jesus Arroyo Torrens. This software performs the calculations to generate the three-dimensional model using the triangulation method, in addition to having a calibration routine [5].

3.2.5 Tools Comparison

Between the low cost options that are possible to be applied in this work, some options were chosen, being the first criterion to verify the feasibility of using each one the acquisition time and acquisition processing time. The Photogrammetry method needs a long acquisition time, since more than 30 photos are needed at different angles of what is

wanted to be acquired, in addition to having a high processing time for these images to finally generate the 3D model, thus it was decided to use the other possibilities.

RealSense has a good resolution and fast acquisition, however, was not used because Intel discontinued the company's own free software, used in previous RealSense D400 (one of the equipment that was available to this work) line to obtain the 3D models. The other software programs available do not export in the free version, or had difficulties to make the acquisition manually, losing the reference point during the acquisition, besides having a very limited maximum scan area.

Ciclop was a application considered due to its simplicity of necessary hardware and also to have a totally free software, that exports totally free and has its own programming and operation documentation available. However, due to COVID-19, it was not possible to continue constructing the hardware, thus the software chosen for the project was Skanect, which uses only one Kinect for the acquisition, has the possibility of free exporting the acquisition with a reduced set of data, and which has a sufficient resolution for the purpose of the project. In Table 3.1 the criteria for choosing the tool are presented.

Table 3.1: Comparison of 3D acquisition options for this work.

Tools	Free export	Acquisition time	Hardware available
Meshroom	✓	✗	-
Kinect - Skanect	✓	✓	✓
RealSense options	✗	-	-
Ciclop - Horus	✓	✓	✗

3.3 Point Cloud and STL

Point Cloud is a three-dimensional points data structure usually used to represent real-world or simulated geometries in the x , y and z coordinates (in case of 3D Point Cloud), and can even represent colours as a fourth dimension, 4D [71].

The data acquired resulting of a 3D scanner must be converted into Cartesian coordinates to make the recognition of object and thus be able to extract information, visualise

and carry out the desired operations for its acquisition, being an essential first step to generate the 3D model [72]–[74].

A Point Cloud is a common representation for 3D scanners and depth cameras, but it can also be generated for depth data simulation programs. There are different ways and programs to deal with the acquired Point Cloud, which depends on the objective of the project. A way to process 3D meshes is with MeshLab, an open-source software that works with Point Cloud and 3D triangular mesh [72], [75].

To use the PC in applications for development of 3D, as 3D printing technology and manufacturing, a CAD software is necessary, which requires to convert the Point Cloud into a CAD model. Currently one of the most widely used formats of a CAD model is STL (STereo Lithography), which is a representation of a 3D surface as a series of small triangles, avoiding large amounts of sampled points in Point Cloud, making a format simple to read and write [76], [77]. Figure 3.10 represents the conversion of Point Cloud to STL format.

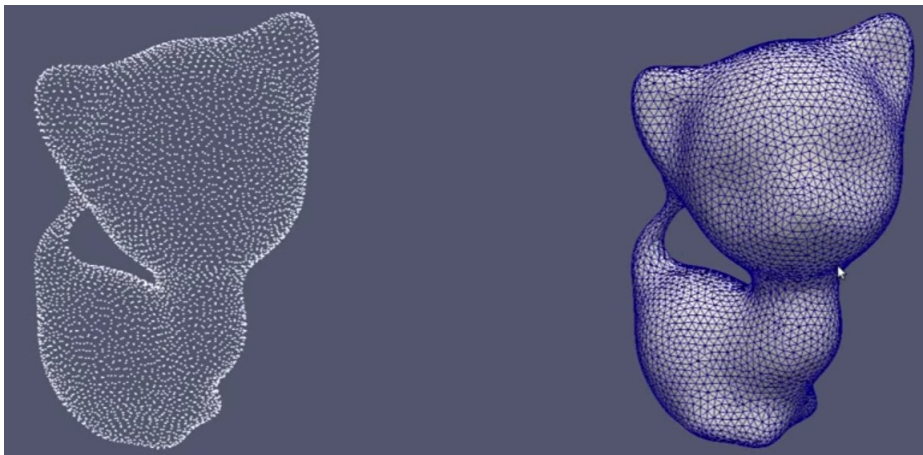


Figure 3.10: Conversion of Point Cloud to STL format. Image from [78]

3.4 Software for Adjusts: MeshLab and Meshmixer

To perform the conversion of the scanner results, which can generate a Point Cloud (.PLY) or even .STL format in which there are important parts with gaps, it should be used a

software to perform treatment and conversion to a solid 3D object, and then read the files into a CAD software [7].

MeshLab is an open-source software able to import the .PLY or .STL file, to clean up and edit existing imperfections due to some flaw or lack of precision in the acquisition process and then convert into a solid object [7]. Another software for 3D scanner post-processing is the Meshmixer, which works with triangular meshes with tools for filling holes, make plane cuts, 3D measurements, reducing or simplifying the mesh, mesh smoothing, orientation optimization and automatic alignment of mesh [79]. The uses of both software programs will be shown in this project in the System Methodology section.

3.5 Machining

The need for greater flexibility to fit the changes that happen in the market and in the global economic environment, generates a development in machining and tool design technology, which reflects in high quality and accuracy in production and lower weight and dimensions in the tools [80]. These facts point to an evolution in manufacturing engineering, indicating a focus for low volume and wide variety of productions, thus requiring high flexibility in manufacturing lines [81], [82].

Machining is the term used to describe various types of operations, which consist of removing material from a workpiece, whether in cutting processes, abrasive finishing processes or unconventional processes (using sources of electrical energy or chemistry). Within cutting operations, turning, drilling and milling are more prominent, and they need a specialized tool or machine to perform each of these processes. Thinking about the industrial sector, robots are used to perform these types of machining tasks, in addition to working as material handling, welding, painting and polishing [36], [83].

Industrial robotics have become an indispensable part of manufacturing as can fulfil the need of the industry in terms of accuracy and flexibility with a reasonable cost and production time [75], [82], [84]. Due to this, the sales of industrial robots is quite promising, having an average growth close to 15% per year between the years 2017 and 2020

[85].

3.5.1 Milling Tools

The machining process is the material amounts remotion in a part to be machined by a tool. There is a wide variety of tools for milling operations, in which mill tools is used to end and roughing process and are usually made of High-Speed Steel, Cobalt Steel, or Carbide, and for longer life, they can also be coated with titanium nitride (TiN). Theses end mills are categorised as Flat, Ball, Bull and Chamfer, which are represented in Figure 3.11 and defined below [86].

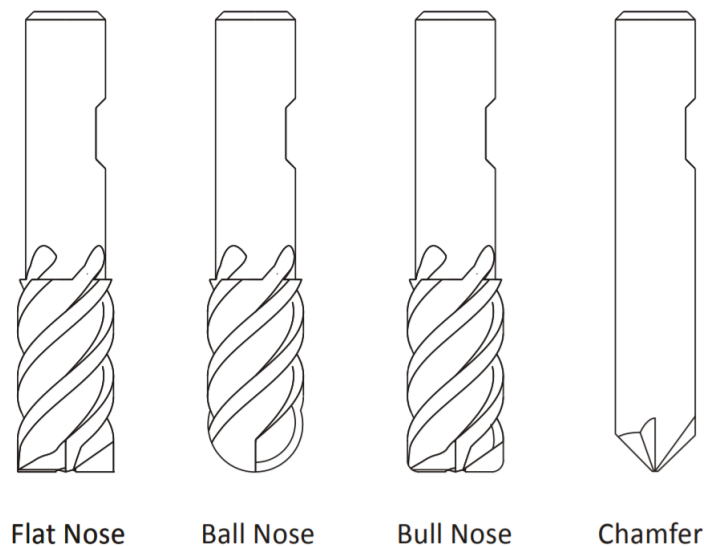


Figure 3.11: Mill tool nose types. Image from [86].

Flat Nose mills are used to 2D mill, contours and pockets.

Ball Nose mills are primary used for 3D milling.

Bull Nose mills are used for roughing operations and by having a radiused corner and a great resistance, allow creating a fillet at the bottom of a wall of the machining material.

Chamfer mills are generally used when finishing or refining the product, by doing a peripheral cut along an edge of the workpiece, such as have an angled nose used to create a chamfer (angled surface) or to de-burr parts.

3.6 Industrial Robots

An Industrial Robot is a manipulator multi-function that performs different works of the human, defined according to ISO 8373:2012 as the combination of manipulator, actuators, controller, teach pendant and a communication interface, which is used in automation applications for multipurpose, being fixed or mobile, that can be automatically controlled, reprogrammable and programmable in three or more axes [13], [87], [88].

A robot is particularly advantageous for industrial applications, such as handling in manufacturing processes, loading and unloading machine tools, welding, painting, assembly and machining. Being a rising technology, the implementation is improving to uses techniques that make robots tasks and its implantation easier, either by software or interactions with the environment in a more practical and quick way, in which CAD and CAM tools are used, besides trying to implementing the robot in the most flexible way possible [88].

3.6.1 Configuration

A typical industrial robot system contains a user interface, in addition to the appropriate actuators and controller, depicted in Figure 3.12, used to make repetitive actions. The typical industrial robot used in industries usually has an anthropomorphic architecture, which is named robot arm due to its appearance with a human arm [36].



Figure 3.12: Typical industrial robot system.

To understand the movement and application of a robot, it is necessary to understand its physical configuration. In addition to having different shapes and sizes, industrial robots can present one of following the six main configuration: Articulated, Cartesian, SCARA, Delta, Polar and Cylindrical. The robotic arm used in this work has an articulated configuration, having a base to fix it on some surface and a sequential connection, where each link is connected by joints, until at the end of the arm where the tool is installed to do the job [89], as shown in the Figure 3.13.

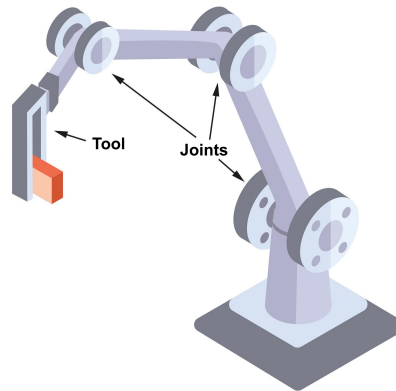


Figure 3.13: Mechanical configuration illustration of an industrial robot [90].

3.6.2 Workspace

All the area and points where the robot end-point can reach is called workspace, that is, places where the robot can perform a task, which in some cases could be the tool or wrist movement [12]. The workspace needed to perform a task implies choosing a robot that meets these requirements, that is, the parts of the robot must have the necessary dimensions to obtain the workspace desired [12], [91]. To better understand Figure 3.14 shows the workspace for the tool and for the robot flange.

It is important to note that to reach some work points it is necessary that the robot has all its joints in certain physical configurations, and depending on the desired movement points, there are more or fewer possibilities of its joints configuration to reach this point. [91].

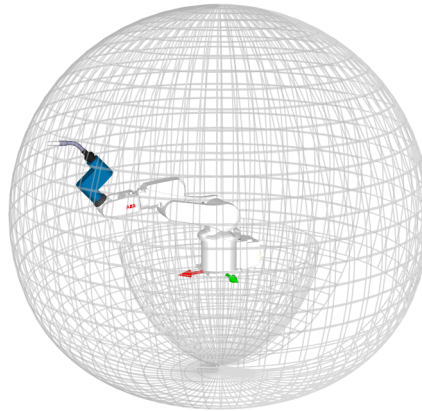


Figure 3.14: Robot workspace illustration.

3.6.3 Robot Path

For a robot to perform its tasks or perform any action, it is necessary for it to make a movement, and with the planning of robot motion makes possible to control it, through inputs in the robot's system controller to it follows the desired trajectory. This trajectory can be defined by a series of points that the tool must pass, and there is an exponential increases in computational complexity with the number of Degrees of Freedom (DOF) [12].

For the execution of a common movement of the robot, inverse kinematics is used, but in addition, it is necessary to consider collisions in these movements, be it between the parts of the robot itself or with objects in the workspace, when is creating the toolpath between the whole desired points that the tool will pass [75].

There are different ways to plan the trajectory of industrial robots, in which these trajectories can be defined by points, that are joined by polynomials or splines. Considering more than five points for the trajectory, it already involves high-order polynomials, which become difficult to predict, so in these cases splines can be used [92].

3.7 CAM Software

A computer software that is used to control machine tools and other machinery aimed at the manufacture of workpieces is classified as CAM, in which each of these software has properties that directly influence how to use the software and what are its functions, in addition to what formats are generated as a response to the environment [93].

These software can create path planning to perform roughing, semi-finishing, finishing and contour milling, thus allowing the creation of pieces machining programs based on geometry definitions and the parameters of the machining operations to be carried out, usually using a G code language [83].

For that, the parameters of the initial part (in dimensions) and the final model to be machined (CAD model) must be defined in the software, choose the tools to use, define the cutting parameters, and the path strategy to be carried out and, among others. That makes it possible to view the piece after each operation and check the quality of the machining process, in addition to detecting possible collisions and the machining time. Finally, the code is generated in the format equivalent to the machine that will execute the machining process [83].

3.7.1 SOLIDWORKS CAM

In 2018 SOLIDWORKS added a new feature (based on CAMWorks from HCL Technologies) to integrate the design and manufacturing process in just one software, called SOLIDWORKS CAM, that is an add-on to all versions of SOLIDWORKS CAD, which allows to integrate the manufacturing process together with the design process, thus avoiding product costs and delays due to evaluate designs earlier to ensure if they can be manufactured. Basically this software uses the CAD models to generate the toolpath of the machining process, as depicted in Figure 3.15, being also capable to change machining strategies according to the tolerances of a given part file, which helps to maintain the programming consistency, saves time using information that is already created in other 3D models, and with the VoloMill technology it is possible for users to increase the output

while extending tool life and reducing machine wear [94].

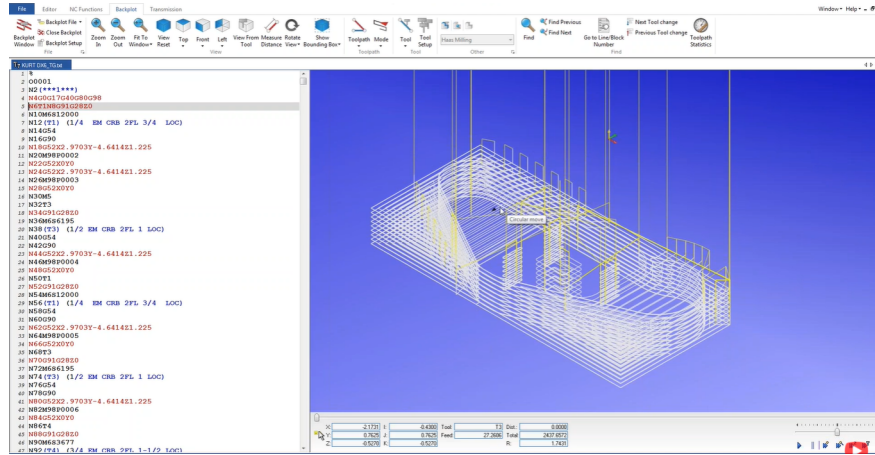


Figure 3.15: Toolpath of milling process generated by SOLIDWORKS CAM. Image from [95].

SOLIDWORKS CAM is available in 4 versions. The first and more simple is the Standard version, which offers the possibility of making configurations for milling without leaving the 3D CAD environment of SolidWorks, which is also present in other versions. The Professional version includes machining assembly, glsHSM, turning and the ability to program four- and five-axis machines. Machinist Standard version allows a part-only modeling environment, where it is possible to work seamlessly with part files imported from several formats. Lastly, Machinist Professional is the most complete version, having all the capability of the Professional version and allows to import and design other manufacturing components used in the machining process, together with a mode that allows to automatically clip the tool path to avoid collisions with custom fixtures or vises [95].

3.7.2 Autodesk Fusion 360

Fusion 360 was introduced in 2016 by Autodesk, and it is named to refer to the various combinations it makes, as the software integrates CAD, CAM, Computer-Aided engineering (CAE), and Printed Circuit Board (PCB), thus being able to do industrial design, structural design, mechanical simulation with rendering and animation, and CAM processes in a single software. In addition, it has tools for collaboration and sharing via the

cloud, so that the design, research and development process takes place only in one software and with access to several sectors of the product under development simultaneously [96].

A prominent feature of Fusion 360 is in relation to the modelling process, it is its ability to be able to view the results throughout the whole process, even of models acquired by other CAD systems. CAM features use HSM technologies that generate toolpaths quickly with reduced cycle times, machine and tool wear and produce high quality parts, working with milling, turning, additive or other manufacturing resources [97]. The milling process is very reconfigurable, being able to import CAD parts and use existing software tools or create one according to the dimensions and model desired, so as to create the tool path, which can be in 2D, 3D, multi axis or to drill, where it has several algorithms for creating the routes.

Autodesk provides a 1-year student license for all of the company's products and services, which includes Fusion 360, and it is possible to renew if the requirements to obtain access continues to be met [97]. More information about the software as well as its interface will be ahead in the next chapter.

3.7.3 RobotStudio

RobotStudio is ABB robot programming software, which is made available by the manufacturer itself, which allows to create virtual environments to simulate real environments, by making available models of robots, tools, positioning tables and other industrial manufacturing resources, all on a real scale. It is also possible to import CAD files, to include previously modelled parts from outside the virtual environment [13].

Being highly focused on ABB robots, it has its own models of virtual controllers, which can simulate programs in a virtual environment, to choose the best configuration for a given robot and check for collisions. The software also allows the conversion of the simulation instructions to the language of the actual controller, RAPID, which can be easily transferred via an Ethernet or USB connection to the controller [13].

RobotStudio has an add-in, the Machining Power Pack, which is dedicated to machining and finishing operations. This add-in allows an easy generation of trajectories, where it is not necessary to select all the various points of the surface that will be worked, as in the normal RobotStudio, but only the surface, being the toolpath are identified and generated automatic for this surface. The toolpath can be implemented normally with a position and speed control or force control, which can control the forward speed according to the registered force [13].

3.7.4 Mastercam

Mastercam is a software for the development of CAM and CAD services, created by CNC Software Inc., which is one of the oldest companies in the manufacturing sector and is the most used CAM software in the world, reaching 27% in 2015 and 22% in 2017 of High-End CAM share [98], [99].

The software works with the creation of CAD models in 2D and 3D, in addition to export these models in different formats, as in SolidWorks and AutoCAD format. Being a software aimed at the creation of the toolpath, either it for milling, lathes, electrical discharge machining, multi-axis machines, wood and art work, having functions to milling, finishing, drilling, contouring, etc [36]. Widely used in the aerospace, automotive, vessel, medical devices and instruments and molding fields, includes Dynamic Motion which in addition to calculating the tool's movement path, adjusts cutting parameters according to the tool's movement, decreasing processing time and tool wear, keeping the tool stable and reducing vibration [100].

It also has a virtual environment for simulation that allows to simulate, in addition to the part and the tool and the machine chosen (in CNC cases), to check the system collisions. Once verified the collisions, it is possible to generate an NC program, which can be exported in various formats, due to the software offering the possibility of including post processors or controllers from other machines, which together with the RobotMaster add-on, allows to make the process machining with robots of various brands and their

specific tool [36].

3.7.5 Software Comparison

In order to choose a CAM software, some commercial software was studied and researched with the necessary criteria to be able to perform the practical work, and software that did not meet these criteria was discarded. First, was searched CAM software applications that makes possible a machining process with robot, in which the software previously mentioned in this section was found. So with these software, other selection criteria were carried out, that can be better viewed with the following Table 3.2.

Table 3.2: CAM software comparison in the project criteria.

Software	License availability	Machining with industrial robots	Machining of complex parts
SOLIDWORKS CAM	✓	✗	-
Autodesk Fusion 360	✓	✓	✓
RobotStudio	✓	✓	✗
Mastercam	✗	-	-

So first a search was made to obtain a license for these software, then follow the search with which license was obtained. In this step, Mastercam was eliminated, being the only one in which a license was not obtained by the tried methods. The next to be discarded was SOLIDWORKS CAM, due to not finding research and videos reporting the use of this together with machining with industrial robots, and then RobotStudio was also discarded since most of its machining applications found to be back to simple and its machining features are very manual compared to similar software features. Thus, the software chosen to continue with the project was Fusion 360, which needed an add-on to work with the machining using industrial robots, but a student license was also obtained for it.

3.8 G-Code and Industrial Robot Code

CNC machines and 3D printers use a programming language to describe axis movements and perform their operations called G-code, which was developed to standardise CNC machine programming [101], [102]. This code contains information for the CNC machine, such as the depth and speed of the cutting movement, rotation speed of the tool, coordinates for the movement of the machine in all its axes, turning resources on and off, what kind of movement to do, among others [7], [103]. All of these commands that the machine must perform are generated in sequence according to the order of operation and are recorded in a text file, which is sent to the machine that will perform the task [7].

The G-code can be generated manually, such as by acquiring real points, moving the tool to acquire the desired coordinates, or obtaining these coordinates through a virtual version of the machine, to then program the tasks by command lines that will be performed [7]. To generate the G-code automatically, it is necessary to use a CAM software, using a 3D part model and a virtual part to be machined to calculate which points the tool will pass through, thus being able to find the toolpath and generate the G-code.

Unlike CNC and 3D printers, that can be programmed in G-code and have much software to generate the toolpath, industrial robots lack this availability, because each brand of the robotic arm has its own robot language, and the software to generate the toolpath is generally of high value, may not cover more than one brand and furthermore, a robotic arm has 4 axes or more, and a G-code usually has an x , y and z -axis as output. A workaround is to use software that generates G-code and then convert it to the programming code of the robot that will perform the machining [38].

RoboDK

To convert the generated code, in addition to the coordinates, the angles of all axes of the robot must be taken into account, even though 3 axes can meet the needs, the others provide greater range flexibility and more possible machining faces. RoboDK is a software which, among other functions, allows the conversion of G-code to coordinates of the robot,

with all angles of axes, being also able to adjust the point of origin of the machining with the reference point of the G-code [38].

RoboDK is a software for industrial robots simulation and programming, also has in its library several robot arms (over 500) of different robot manufacturers (more than 40); Figure 3.16 shows the variety of robots in relation to the number of axes, brand and weight. It is possible to import the tool itself from a CAD model and use one of these robots in the library to the machining process, like a milling machine (as CNC) or a 3D printer converting the NC or G-code programs to robot programs [104].

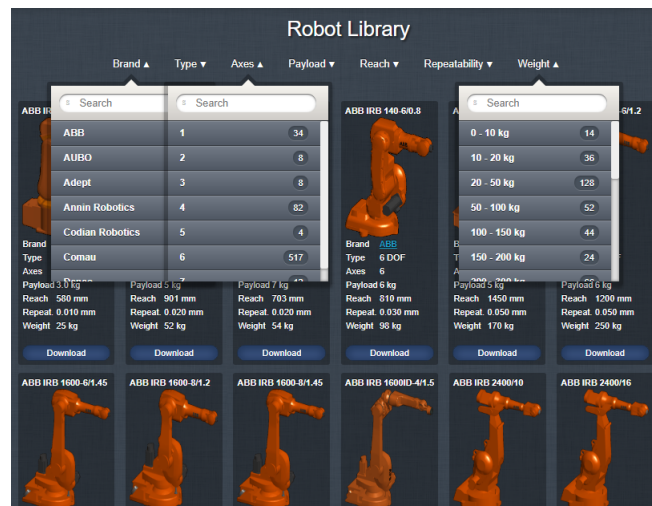


Figure 3.16: RoboDK library showing the variety of robots in relation to the number of axes, brand and weight.

RoboDK will also automatically optimize the robot path, avoiding collisions, singularities and axis limits with high accuracy, that can be calibrated and tested. With the simulation and offline programming tools, after creating a virtual environment, is possible to program robots directly from the computer and outside the production environment, without learning robot-specific programming, reducing the production time. In addition to make possible uses different robots for various application, such as machining, welding, cutting, painting, inspection, deburring, among others [104].

Chapter 4

System Methodology

This chapter is dedicated to addressing the challenge of this work, being discussed in the first Section the existing problem and the basic idea of the solution, in the following Section approaches the proposed solution, such as the tools and software used, and in the last Section the development necessary for the interconnection of the parts of the solution, allowing to execute the whole solution.

4.1 Problem Description

The process for the reconstruction of something or an object that already exists in the real world, is in constant development, thus seeking, more and more, that this process occurs more easily and viable for the various applications, occurring since engineering areas, even for medicine and conservation of historical monuments.

The need to recreate an object can occur for several reasons, such as a product no longer being manufactured, loss of the original design for construction, improving an existing product, analysing competitor products, conserving a model or sculpture, resizing a model, reproduction of an organ and among other applications.

To facilitate the process of recreating an object, 3D scanners emerged, which assist in the remaking of 3D projects of great complexity, thus facilitating the modelling of 3D objects that can be used as previously mentioned. More precisely, some examples are: 3D

printing; simulations of real equipment; creating parts and objects in games.

After obtaining a virtual 3D model by a 3D scanner, it is necessary to use a machining process, such as CNC, or a additive manufacturing, through 3D printing. However, due to the possibility of different materials used in 3D printing be limited and that CNC machines have limited working area, and are targeted only to machining operations, the idea of using an industrial robot in the machining process arises.

Industrial robots have low rigidity compared to a CNC machine, which generates vibrations and, consequently, deformations, in addition to the fact that most commercial software for toolpath is aimed at CNC machining, both for the creation of toolpath and for the configuration of the machine characteristics. However, industrial robots also have their advantages, such as their large work volume, lowest initial cost and multitasking ability, which allows high adaptability, of great value for manufacturing, since the market is constantly changing and needs to produce customised parts.

Currently, companies such as ABB, FANUC, KUKA and Stäubli are investing in industrial robots for machining operations, each one already having a robot model focused on this task, with high rigidity and better repeatability [36]. In addition, as mentioned in Chapter 3, CAD/CAM software is implementing the functionality for converting G-code to robot codes.

Thus, the objective, as already described in Section 1.2, is to acquire a real shape or object for the virtual world through a 3D scanner, and use this model to carry out machining of materials (such as polystyrene, polyurethane, or other low rigidity material) using industrial robots, with the toolpath obtained through CAD/CAM programs. These processes can be viewed in Figure 4.1 bellow.

4.2 Proposed Solution

To accomplish the objectives of this work, of machining parts acquired by a 3D scanner, will be used the methodology shown in the block diagram of Figure 4.2. This methodology uses the Skanect software to acquire the 3D image with the Intel Kinect, adopts the Fusion

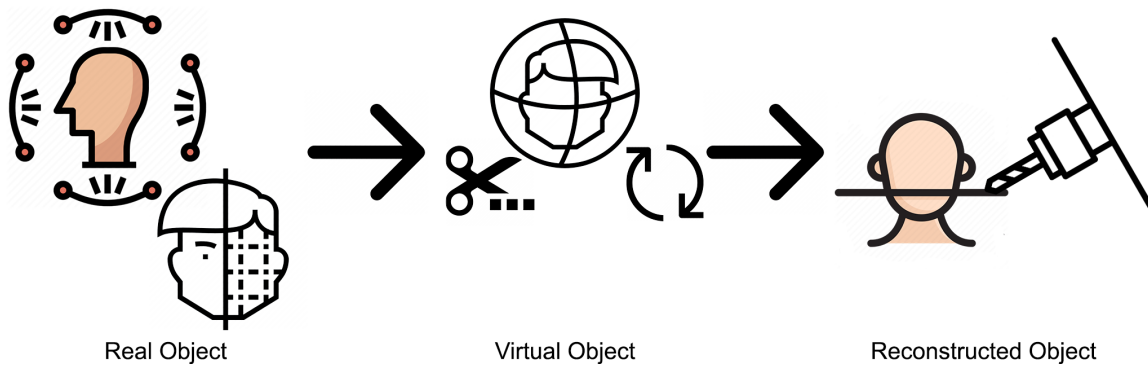


Figure 4.1: Project architecture.

360 to generate the NC code, and RoboDK to convert the code to the ABB2600 robot with the IRC5 controller in iiLab of INESC TEC. Each of these tools and steps will be explained further below, in this and Section 4.3.

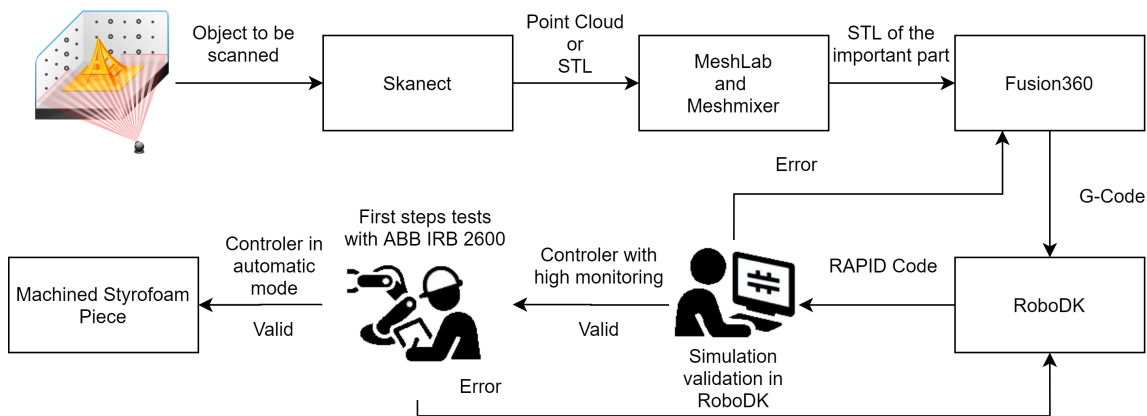


Figure 4.2: System architecture block diagram.

In the first stage, the object that will be acquired in 3D must be positioned in a place that allows to perform the acquisition movement around it or on a platform/chair swivel. With the object properly positioned, the Skaneect software makes the acquisition and exports in the STL format. Then, this STL must be treated to become a solid object in the MeshLab software and is cut only the part of interest in Meshmixer. The edited STL is imported into Fusion 360, which uses several tools to create a toolpath for machining a workpiece in G-Code format. An add-on in Fusion 360, exports that G-code

to RoboDK, which will convert it into a RAPID code to work with ABB robots, using specifically the IRC5 controller. Finally, with the generated RAPID code, simulations are performed inside the RoboDK itself, to validate the toolpath and machining position, and then tested on the ABB 2600 industrial robot. First is checked whether the initial steps of the toolpath go as expected, and if so, the workpiece is placed to be machined automatically, following all the lines programmed by the software.

4.3 Solutions Development

This section deals with the adaptations, tests, procedures and tools used to achieve the objectives of this work. Being the first objective the acquisition of the 3D model, and in this stage were tested the possibilities mentioned in Section 3.2, between them Intel Kinect using Skanect (Section 3.2.2) and Ciclop with Horus (Section 3.2.4). Following by the robot and its respective controller and the tool used, the created virtual workspace and the adaptation part to use the tool in a different robot model.

4.3.1 Ciclop

Ciclop it is a low cost options for 3D scanner, both in hardware and in software, which have a free export option, and was aimed at making the acquisition of small objects. It obtains the points of the object through the triangulation method, using a webcam, two laser emitters and a stepper motor connected to a platform to rotate and make it possible to obtain the points of the object. The first step was to create the 3D model of the platform with the stepper motor connection, as depicted in Figure 4.3, and use its 3D printing for tests.

When testing the rotation of the platform with a stepper motor, there was a need to use a motor driver to reduce vibrations, which were displacing objects from the starting point while rotating the platform. Thus, the L298N motor driver was used for the stepper motor, and new tests together with Arduino, as shown in Figure 4.4, showed a reduction in vibration.

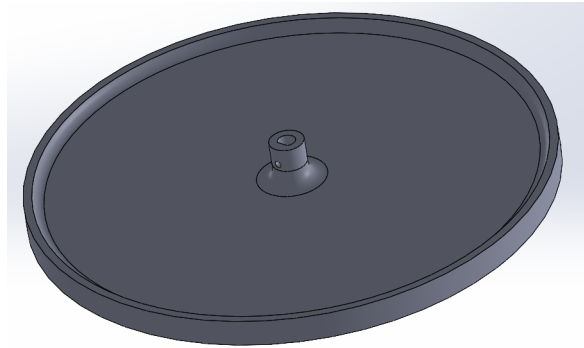


Figure 4.3: Rotating platform for stepper motor.

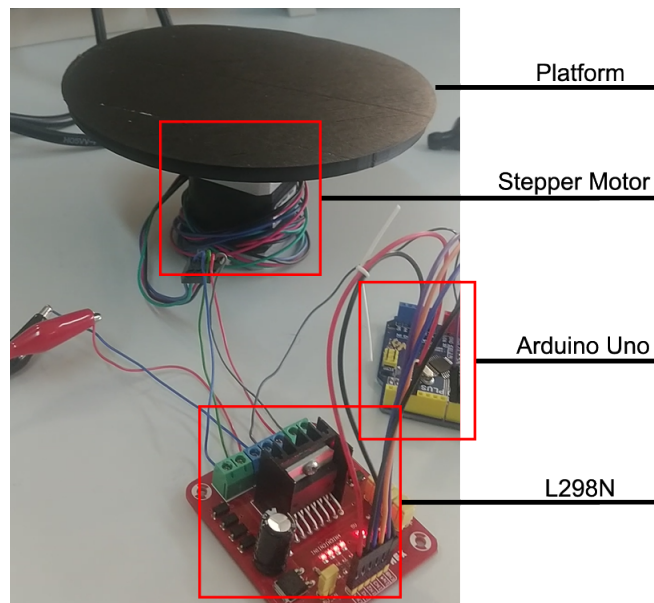


Figure 4.4: Tests with stepper motor controller.

The next steps would be to create a structure for fixing the laser emitters and the webcam, and carry out the 3D acquisition tests, starting with its calibration, using the chess pattern provided by Ciclop. However, due to COVID-19, laboratory tests were interrupted and the use of Ciclop at this work was cancelled, leaving the 3D acquisition to be done only by Skanect. It is important to note that Ciclop exports the generated file as a point cloud (.PLY), that needs to be converted into a mesh file (such as .STL), shown previously in Figure 3.10.

4.3.2 IRB 2600 and IRC5 controller

To carry out the practical tests of machining the parts, the industrial robot IRB 2600-20/1.65 (Figure 4.5) was used. This was available at iiLab of INESC TEC. This is a mid range robot, optimised for applications such as welding, material handling and machine tending, with seven versions available. The version used has a payload of 20kg and a reach of 1.65 m. And aims through process speeds and scrap rates to help increase the output in process applications [105].



Figure 4.5: ABB IRB 2600-20/1.6 industrial robot.

The IRC5 (Figure 4.6) it was the controller used in this work. This controller has the QuickMove and TrueMove functions, aimed to safety, speed, accuracy, compatibility, and reduction in cycle time. As an ABB system, it is programmed with the RAPID

language, which responds to all robot systems of the brand, with its flexible and high-level programming language. It also has a MultiMove configuration, which allows to control up to four robots, and perform complex motion patterns perfectly [106].



Figure 4.6: ABB IRC5 single cabinet controller [106].

4.3.3 Tool Used

The choice of a machining tool to perform the machining processes is very important for CNC machines, and even more important for robotic machining, which has a lower stiffness level and maximum payload, that can limit the possible choices of high speed rotary tools. The spindle DREMEL 3000 tool was an alternative found. It has a weight of 0.55 kg and an adjustable rotation variation between ten and thirty thousand revolutions per minute (RPM). To use this tool it is necessary a support and fix it to the robot. The fixation project was done before by Almeida, using in his work the version of the DREMEL tool and the support modeled for the ABB IRB 140 as depicted in Figure 4.7 [8], [107].

Another step for the machining process is to choose the milling cutter according to the spindle used, which can limit the speed, kind of activities, type of material and the dimensions the roughing material, and define the toolpath generated by the CAM software. The milling cutters available for the spindle used in this work (DREMEL 3000), and that are compatible with the chosen spindle, are also from the DREMEL brand, being



Figure 4.7: Dremel and support for fixing the tool to the robot [8], [107].

a kit that can be used for machining work on soft materials and metals, such as wood, plastic, aluminium, copper and brass [108]. This kit has three milling cutters shown in Figure 4.8, along with the corresponding virtual models (from Almeida) to be used as a basis for the solution simulation, with 2.5 cm of length after the end of the tool [8].

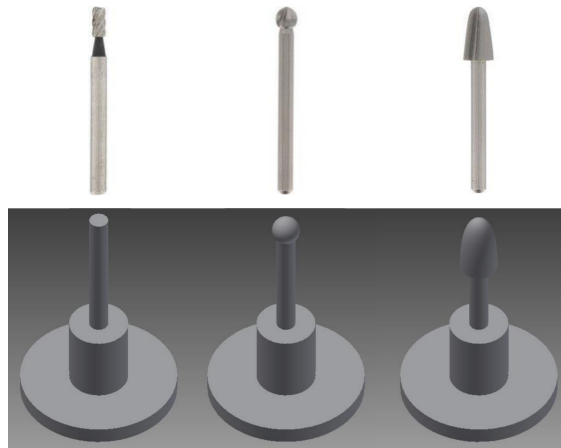


Figure 4.8: Milling cutter kit and its virtual models [8], [108].

To use the milling cutter chosen in the creation of the toolpath on Fusion 360, the data of the respective milling cutter must be imported into the software. That for the process used in this work it was chosen to work with the Flat Nose type (as detailed in Section 3.5), corresponding to the left one in Figure 4.8. The data of the cutter must be entered at the moment of choosing the milling cutter used to perform the machining process, creating the Flat end mill tool as shown in Figure 4.9.

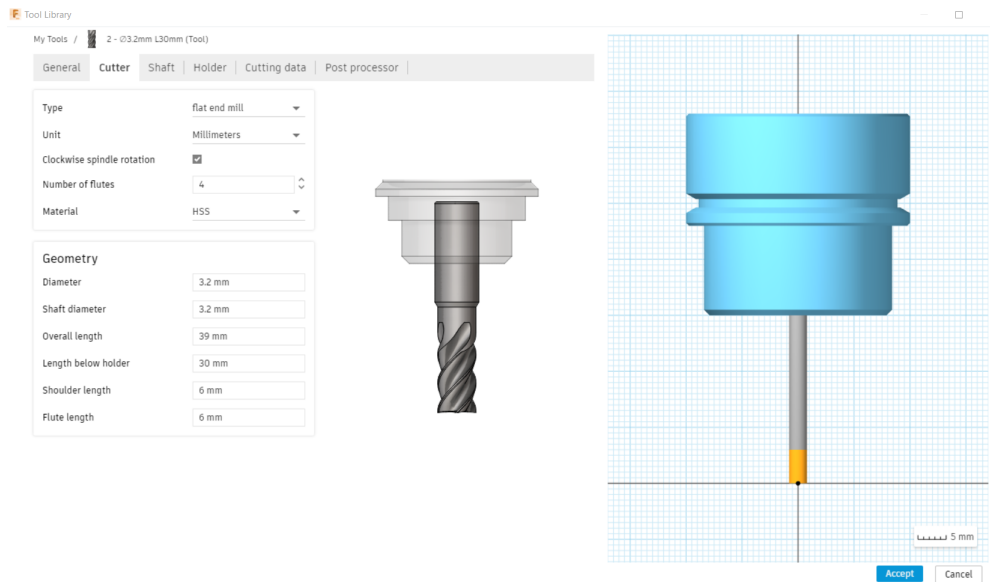


Figure 4.9: Creation of the milling cutter used in Fusion 360.

The use of this tool in RoboDK was merely illustrative in order to have a better visualisation of the simulation, since what the software really considers is the frame of the tool tip in the workspace when converting the input toolpath to the robot's toolpath. Even so, the tool together with the respective milling cutter chosen for the project, designed by Almeida, was imported into the software, and the its attachment to the robot is as shown in Figure 4.10, where it is also possible to see the real tip of the tool as the frame just below the tool.

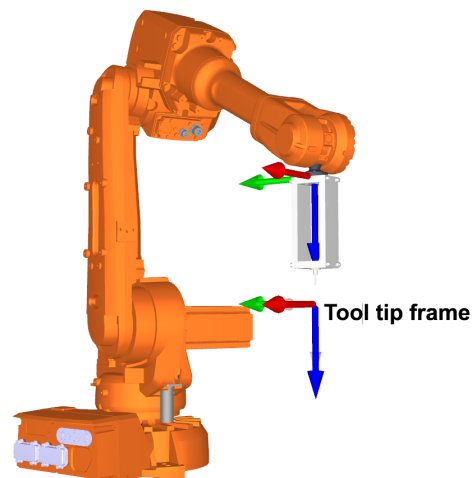


Figure 4.10: ABB IRB 2600 with the respective tool frame.

4.3.4 Virtual Workspace

The work environment used in iiLab did not contain possible collisions around the robot or with a near range to the machining site and, in addition, the machining process was always tested previously in software to verify the possible occurrence of collisions, and was always supervised in practice. Considering this, the virtual workspace created included only the robot IRB 2600-20/1.65, the tool model, a part to represent the workpiece, the respective tool tip frame (which is calibrated with the real values) and a reference frame to the machining location, represented in Figure 4.11.

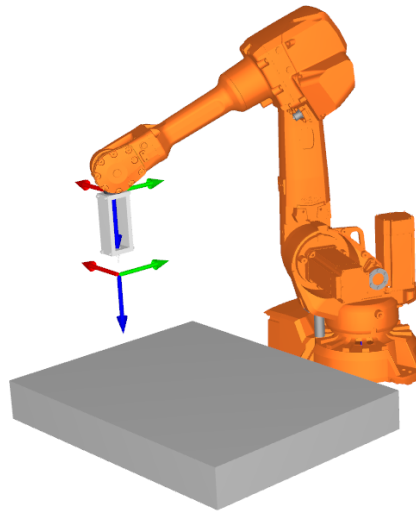


Figure 4.11: Virtual workspace in RoboDK.

4.3.5 Tool of IRB140 to IR2600

The support model for attaching the tool to the robot was designed by Almeida to be used together with the ABB robot IRB 140. As in this project the robot used was the ABB IRB 2600, it was necessary to create a part to allow attaching the tool support to the robot. Thus, using ABB's own library of its respective robots, it was designed the junction part, according to Figure 4.12, which was screwed with two M6 screws to the IRB 2600 Tool flange, and to the tool support with 4 M6 screws.

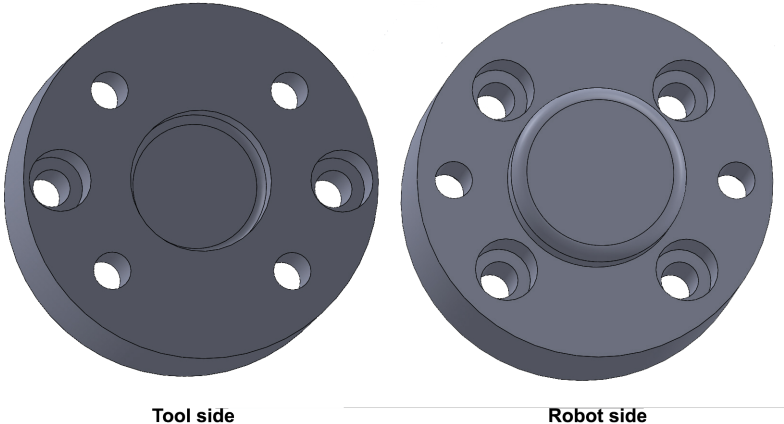


Figure 4.12: Design model of the junction part.

With the part properly modelled, it was 3D printed and together with the tool, fixed to the robot, as depicted in Figure 4.13.



Figure 4.13: Tool fixed to the IRB 2600 robot using the junction part.

Chapter 5

Development and Results

This chapter deals with the implementations and results of the work. The first section addresses how the object acquisition and reconstruction process were carried out, followed by the types of tests performed and the results and comments of these tests in the next sections.

5.1 Implementation

The first stage of the project is the 3D image acquisition process. For this work it was decided to carry out the acquisition of people's faces using Intel Kinect with the Skanect software, as discussed in Section 3.2. These acquisitions can be carried out with the part to be acquired static and by moving the Kinect around to get the points of interest, or with the Kinect itself fixed and rotating the part to be acquired. In both cases, it is needed to accomplish an acquisition without sudden movements and maintaining a certain distance (the software itself indicates if it is correct) during the acquisition, so as not to lose the reference points or make wrong acquisitions. If errors occur in the final model, it is necessary to make another acquisition.

5.1.1 Acquisition Steps up to .STL Format

When starting the acquisition, a cube is generated with the volume to be acquired. Parts outside this cube will be ignored in the acquisition, and is shown inside of that cube a preview of the 3D acquisition. The right side of the screen shows the real-time image of the Kinect camera and depth camera, along with a screen to indicate the distance between the object and the Kinect during acquisition, and the green colour indicates the part that will be acquired, as shown in Figure 5.1.



Figure 5.1: 3D acquisition process using Skanect.

The result of the acquisition is processed by the software and can then be post-processed on itself using tools such as to reduce the number of faces of the scan, look for places with holes and attempt to fill them, and move, crop and remove random or floating smaller parts. Finally, Skanect allows exporting the generated file in .PLY, .OBJ, .STL or .VRML format, with a maximum limit of 5000 meshes for the free version of the program.

With the generated STL file of the face, still need to make some adjustments to work on it, using the software MeshLab and Meshmixer. First, the MeshLab functions are used to clean up unwanted points and parts of the acquisition and also to build an entire object

from the STL file. After this step, the new file is treated in Meshmixer to make an object solid, perform cuts, rotate, translate, adjust and resizing of the mesh.

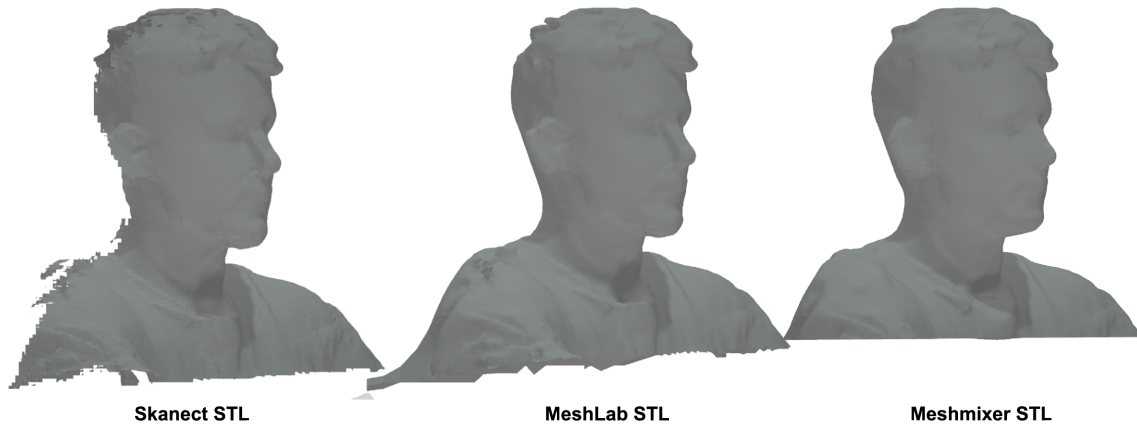


Figure 5.2: Post-processing of .STL with MeshLab and Meshmixer.

In order to reach the model that will be used in the machining process, the characteristics that would limit the milling process in practice were taken into account, avoiding the collision between the spindle support and the workpiece. To avoid this collision, it was required to limit the maximum height of the part to be machined, in order for the spindle support not to pass in parts of the workpiece that should not be roughed out. Figure 5.3 shows the process to reach the interesting part for machining.

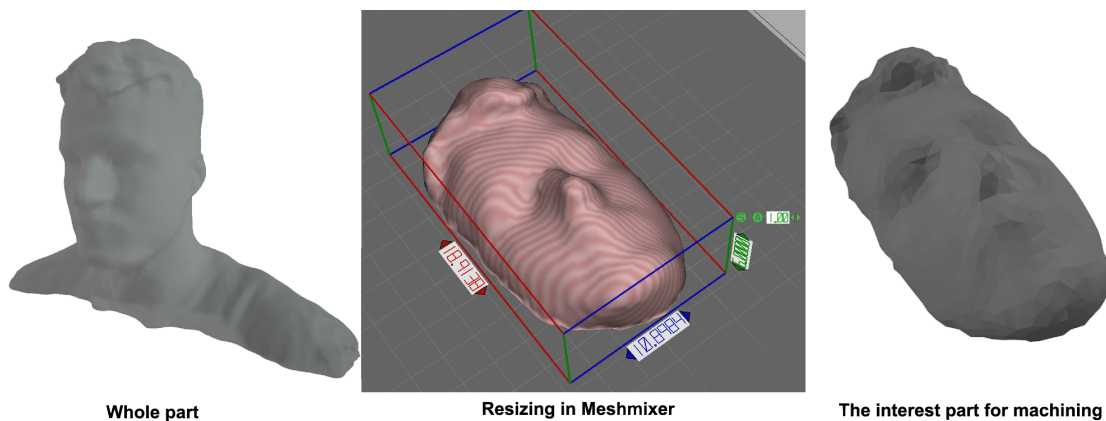


Figure 5.3: Resize of the whole piece to the interest part for machining.

Of all the CAM software studied in this work, the ones available for tests were SolidWorks, RobotStudio, Fusion 360 and RoboDK, which obtained a student license, however,

the most used in the work was Fusion 360 in conjunction with RoboDK.

5.1.2 Fusion 360

With the file of the part for machining ready, it is imported to the CAD/CAM Fusion 360 software. To be possible to work with the STL file in the Manufacture part of the software, first, it is necessary that in the design part of the software is performed a conversion of mesh to a Boundary representation (Mesh to BRep function) creating a new component; depending on the number of the meshes (more than 20.000 "faces"), it might be necessary to previously perform a mesh reduction. Figure 5.4 is an example depicting the conversion to BRep (in this case due to has 2858 "faces", the mesh reduction was not necessary).

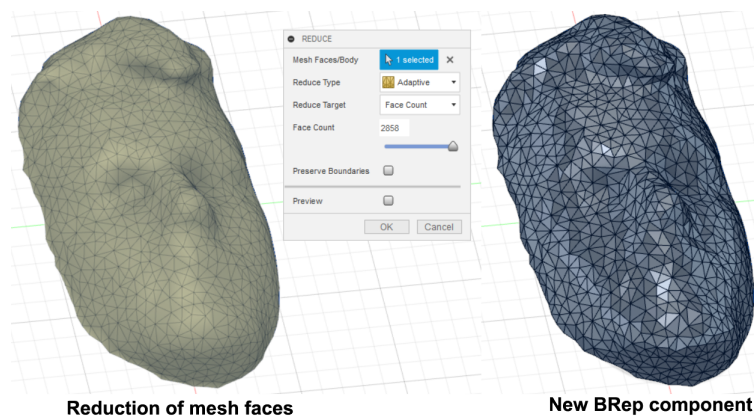


Figure 5.4: Reduction of mesh faces and converting to a boundary representation in Fusion 360.

Before choosing the machining strategy, it is still necessary to create the tool that will perform the machining (as explained in the previous chapter), create a setup that will be considered the total volume of the part to be machined and a reference point in the workpiece. For this setup, the height was considered 1 mm of above the height of the face to be machined, so that even if the actual workpiece has unevenness on the surface, the machining process can thin out the highest points of the part and ensure that the whole face is within the actual workpiece. The point for the origin of the work coordinate system, as shown in Figure 5.5, should be positioned at the upper surface of the part, in order to simplify its use, as will be described in the next steps.

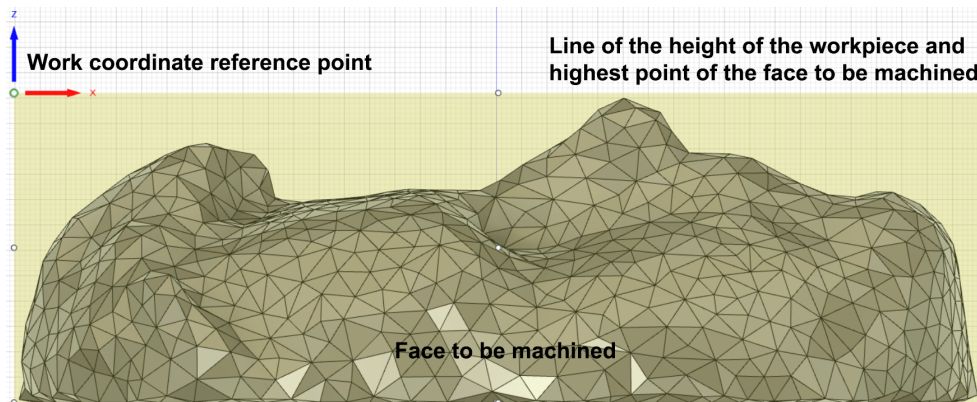


Figure 5.5: Height of the workpiece and work coordinate reference point in Fusion 360.

With the setup and tool created (described in the previous chapter), it is possible to generate the machining toolpath. Fusion offers several strategies for this process, that must be chosen according to the type of format of the part to be machined. Considering that for this project the shape of the part does not follow a pattern, it is necessary to test the machining strategy.

First it is needed to choose between the two 3D roughing options which the software provides, the Adaptive Clearing and the Pocket Clearing strategy. Taking into account that the first option assures the reduction by 40 %, or more, of the machining time, this was the method chosen for roughing.

The toolpath strategy chosen is generated by the software itself according to the settings set at the time of its creation and the shape of the part to be machined. The Adaptive Clearing performs the roughing in levels, starting from the external end of the part and going to the center of the part, making a contour of the object that is on a higher level than the stepdown. Then, make a fine steps upwards in the direction of the tool and make a new contour, repeating this process until finishing the roughing of this layer, as depicted in Figure 5.6.

The maximum roughing stepdown is the working height of the cutter. Therefore, is possible to stepdown more per layer according to this height and reduce the number of layers required to roughing the whole part. In Figure 5.7 a 30 mm roughing was performed by a cutter with a stipulated working height of 6 mm, thus needing 5 layers

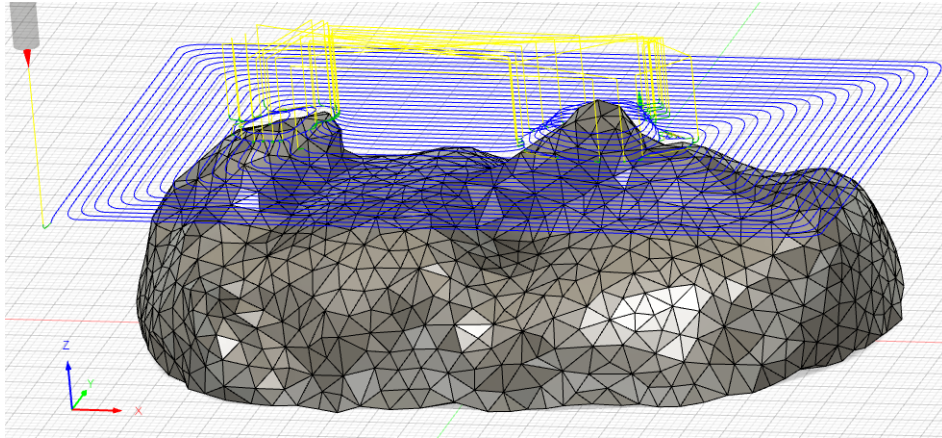


Figure 5.6: First layer of a toolpath using Adaptive Clearing in a face.

for the roughing process.

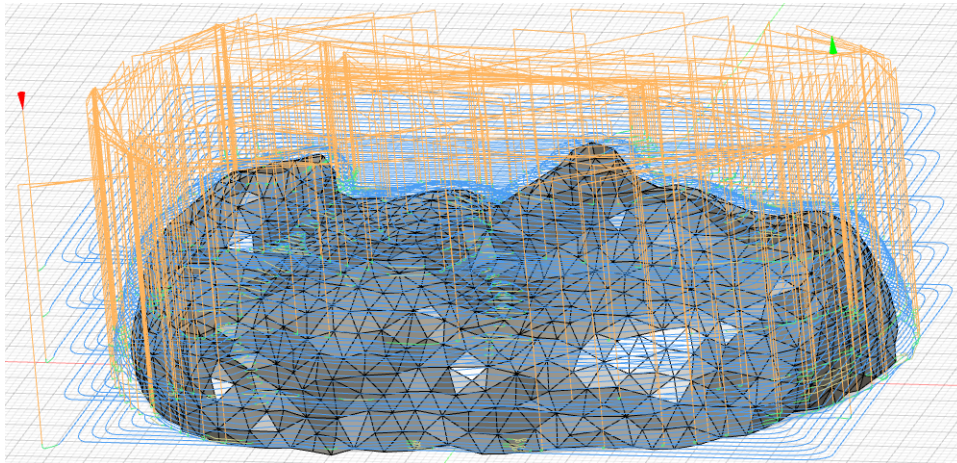


Figure 5.7: Toolpath of a face using Adaptive Clearing in Fusion 360.

After roughing it is possible to give a better finishing to the part with other finishing strategies and using another cutter, in order to get more closer to the desired face model.

5.1.3 RoboDK and Real-World Adjustments

With the creation of the desired toolpath and the RoboDK add-on, in the section tools of Fusion 360, it is possible to send the G-code created for this toolpath directly to the RoboDK software. Before sending the code, the virtual workspace must be configured according to the tools used, as mentioned in the previous chapter, and also with a work

coordinate reference frame, which will be the point that is known in both software. After importing the code, it must to be configured the robot to use, which will be the frame of reference and which tool is going to be used. In addition, it is also recommended to remove the program events, limit the maximum ranges of motion of the robot axes (to avoid collisions) and keep the Approach, Retract and the other standard data generated by the software, exemplified in Figure 5.8. After that, it is just needed to click in the update button to generate the first simulation tests, with the respective toolpath, tool and reference frame.

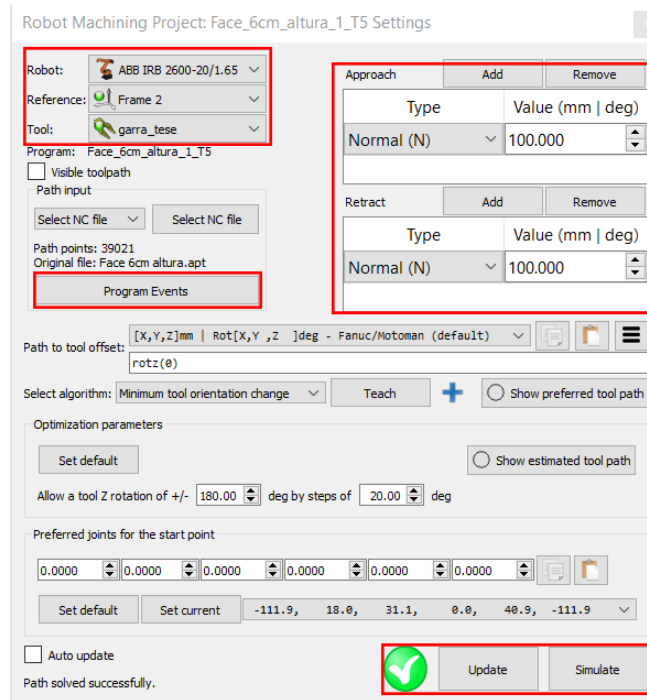


Figure 5.8: Configurations of the code in RoboDK.

In order to perform the machining in practice, the Tool Center Point (TCP) of the robot must be obtained, as this is considered when converting the toolpath in RoboDK. The process of acquiring the TCP is done by seeing its value in the IRC5 robot controller.

To define the tool TCP, four different points must be acquired, as it already provides sufficient accuracy to accomplish the purpose of this work. Each acquired point must be as close as possible to a real reference point, just changing the robot's orientation. Thus, a pencil was positioned to be used as a reference point, and the four points were acquired

with different tool orientations, as shown in Figure 5.9.

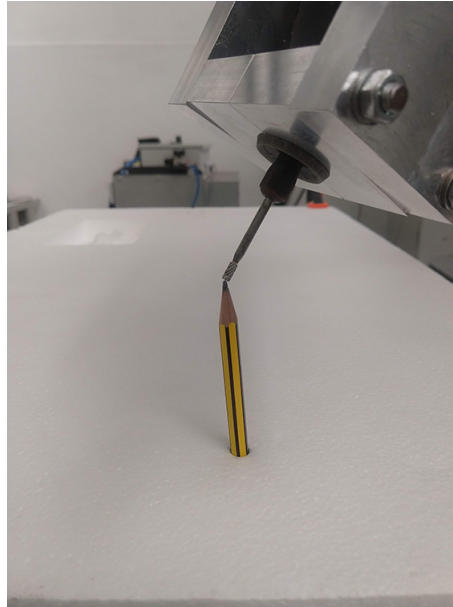


Figure 5.9: Teaching points to perform the TCP calculation.

With the points acquired, the controller performs the necessary calculations and finds the x , y and z position of TCP. In addition, it is necessary to add the tool mass (was used a value of 1 kg, and the value of its Center of gravity (COG), and in this case the most important is the offset on the z axis in which an estimated value of 150 mm was placed, due to the total length of the tool being approximately 300 mm).

With the TCP found, the position desired to start the machining process, which is the same point as the work coordinate reference frame, should be checked. This point makes the connection between a known point in the real and virtual worlds. Once the TCP and the reference coordinate work point are acquired, both must be updated in the RoboDK, then making another update of the generated toolpath, which will be validated with the actual tool and the start machining point coordinate. Figure 5.10 shows the TCP referring to the robot flange and the start frame machining coordinates in relation to the center of the robot, updated with the values obtained by the IRC5 controller. It is important to mention that if the tool is removed of the robot or has milling cutter changes, it is necessary to repeat the TCP definition process.

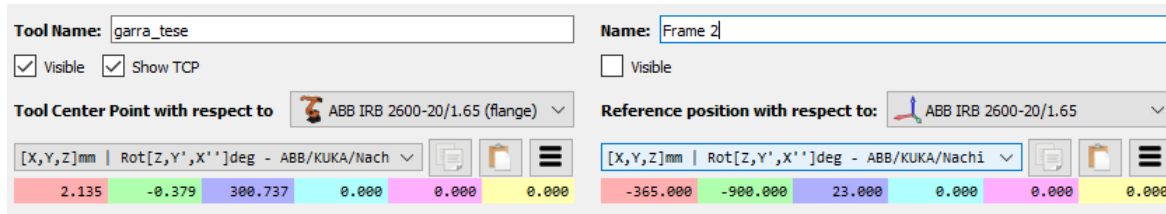


Figure 5.10: Adding the real TCP and work coordinate frame.

With the robot code updated, the simulation will indicate an estimated machining time, and in addition it can finally export the code to be downloaded to the real world robot. The export file will be generated in the ".mod" format. The exported code must be edited manually if more than one .mod file is generated, as only the first file will be opened on the controller. The others will be opened by functions within it, which must have the correct directory path for the other files. In addition, the controller identifies only .pfg files as the programs to load into the robot and, for this reason, it was necessary to create a standard .pfg file, just for calling the first .mod file generated by RoboDK. Figure 5.11 shows the directory path change and the standard .pfg file for running the program "BOT" utilised in tests.

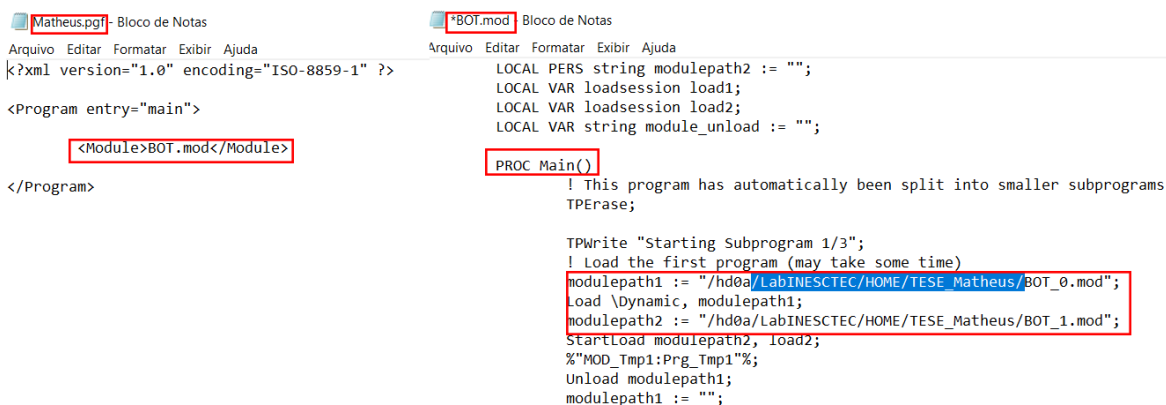


Figure 5.11: Creating a file to open the program and changing the directory path to the correct one.

Finally, to transfer the RAPID code to the robot controller, an Ethernet connection was used between the controller and the computer. Through the FileZilla program the connection to the controller Internet Protocol (IP) address was made, thus enabling the

transfer of files, and visualise the path for the files, to use in the code as previously mentioned.

With the files transferred, the program in the controller is executed. Initially is used the manual mode to check if the robot program is running as expected, requiring the user to click on the button "play" to proceed on each step. Once the program is working as expected, the tool is raised of the part and the controller is changed to automatic mode. Pressing the button to release this mode and running the program again, the robot starts where it left off with more speed.

5.2 Tests Performed

The testing process was divided into four stages: 3D images acquisition, simulations and creation of toolpath in CAM software, simulation of toolpath with robot and the real machining of the part, as depicted in Figure 5.12

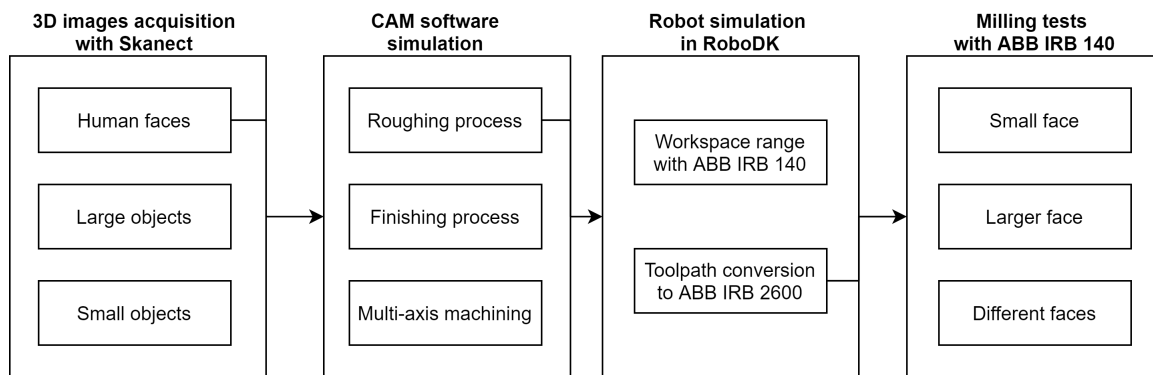


Figure 5.12: Implemented tests block diagram.

In the first stage, was carried out the acquisition of human faces, large objects and small objects with Skanect. Between the results of 3D acquisitions, only the faces acquired were used in the following tests.

The objective of the CAM software simulation step was to verify the influence of the milling cutters characteristics and the strategies chosen, such as the number of stages required to obtain the result and a preview of the result according to the choices, on the

machining result.

Afterwards, tests were carried out in the RoboDK simulation software, to check the workspace range with a different robot (ABB IRB 140) in the machining process, estimate machining time, check collisions and also if the toolpath is feasible for a specific work coordinate point even within the workspace. In addition, to make it possible to see if the machining starting point is in the correct relation to the robot as planned.

The machining tests carried out in the laboratory were done in four days, involving different tests. The first test was directed to a small part within the height limit related to the tool. Then, the next tests were for the same part in larger size, dividing into two stages of machining, one for the top of the part and another for the bottom, thus being able to be twice as high. At this stage the robot's maximum acceleration control was implemented and used in the following tests, using a different face in the machining process and using a different cutter for this same face .

5.3 Results Obtained

The results of this work were divided between the steps of the performed tests, thus having results of 3D images acquisition, simulations and creation of toolpath in the CAM software, simulation of toolpath with robot and the real machining of the part.

5.3.1 3D Scanner

In the first step, tests were performed to verify the accuracy of 3D image acquisition according to object sizes. For large objects were captured a chair and a table with objects on top. Figure 5.13 represents the results of the tests of the acquisition of these objects.

To carry out the test for the acquisition of small objects, a puppet about 27 cm high was chosen. Figure 5.14 presents the result of its acquisition (before exporting the file, without any reduction) and a photo of the original object.

With these tests, it was observed that the 3D acquisition method for small objects lost many details, even though forming a similar structure. Thus, the tests continued

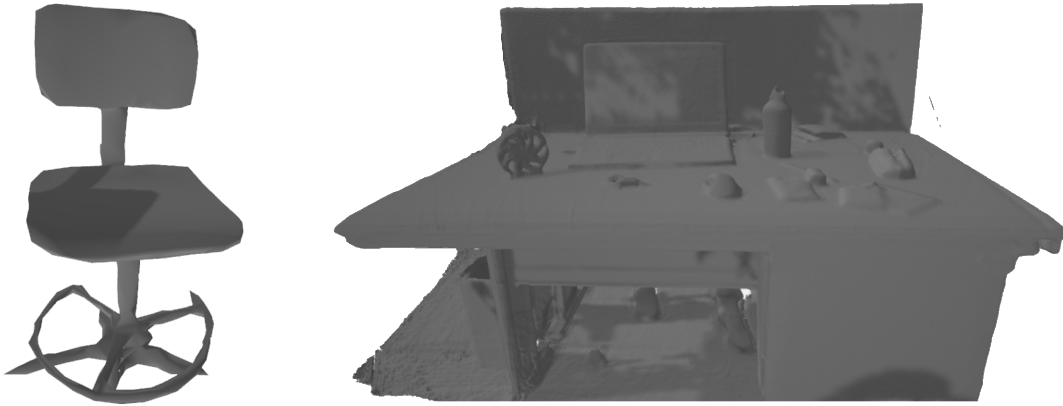


Figure 5.13: 3D acquisition of a chair and a study table.

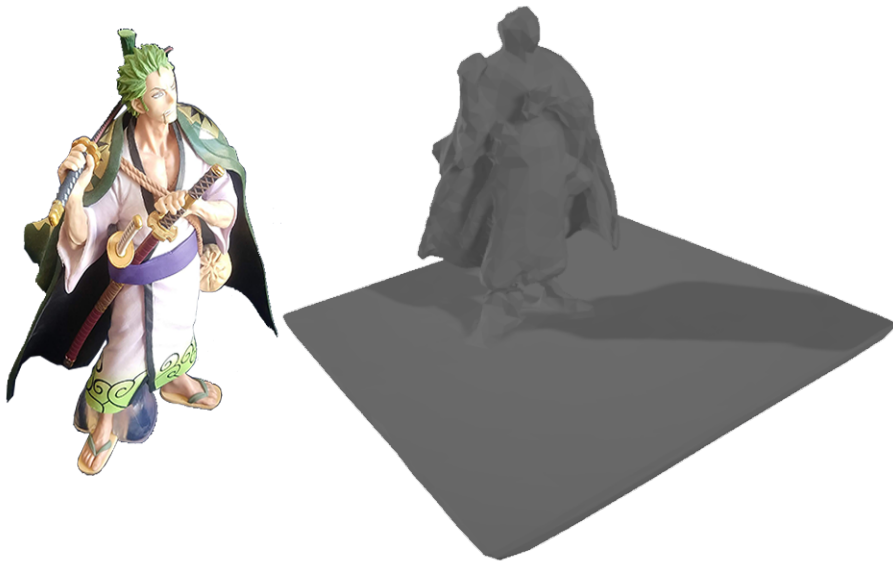


Figure 5.14: Small object 3D acquisition.

performing the acquisition of faces, which have a larger size with details, acquiring the faces shown in Figure 5.15.

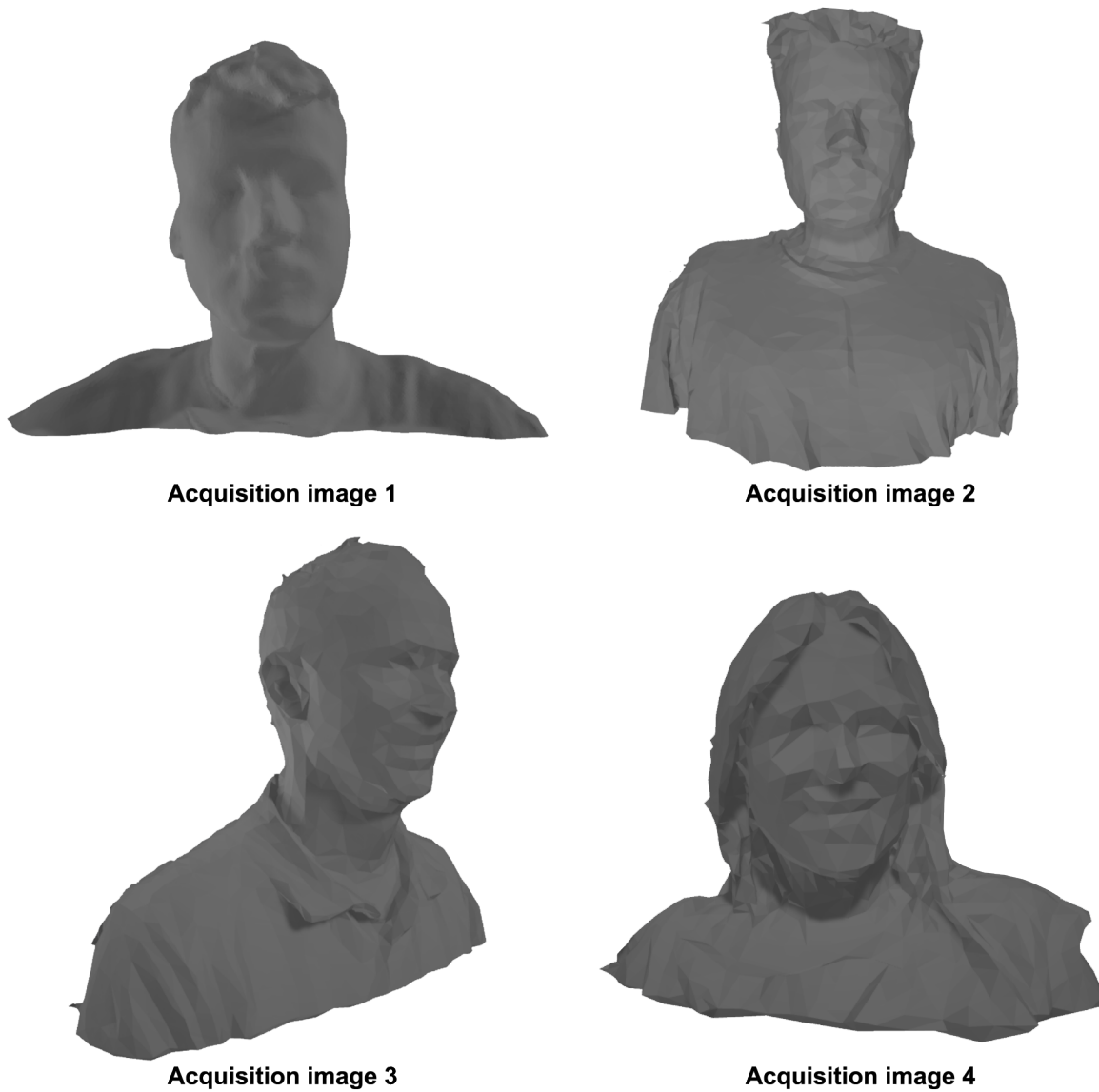


Figure 5.15: 3D faces acquisition.

From these acquired faces, only image 1 and 4 of Figure 5.15 were used in the following steps of the tests. A problem found during this process was that of carrying out an acquisition of a face with a lot of details, such as the case of a person having curly hair. Even during the acquisition process it had a considerable result (similar to other

acquisitions), when it is reduced to five thousand meshes (limitation of free software version) to export in .STL file, a bad result is obtained, as in the case of Figure 5.16.

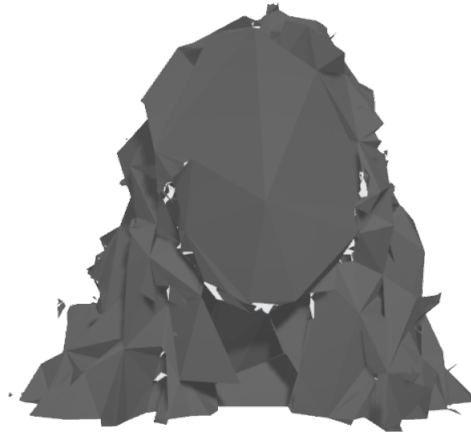


Figure 5.16: 3D face acquisition with lots of details after mesh reduction.

5.3.2 CAM Simulation

With the faces chosen for machining, the tests and steps mentioned in Section 5.1 continued in the CAM software. Roughing was done with Adaptive Clearing and together with it were tested other 3D finishing strategies of Fusion 360. Among those, the most feasible to apply with the geometric characteristics of the faces were Morphed Spiral, Scallop, Parallel and Spiral. A comparison was made between these strategies, as depicted in Figure 5.17.

Among these finishing milling strategies, Scallop works in a larger area of the part and reaches more points (noted in the diagonal view). However, Morphed Spiral and Parallel seems to make the layer connections more smoothly, with great result in the smoothing of the layers, that can be seen in the profile view of the face (especially on the nose part).

In addition to the milling tests with the normal 3D strategies, the Flow strategy was tested in multi-axis mode which allows the machining process to occur on more axes, thus achieving better precision. However, this method does not applied to the BRep format generated, needing to make a mesh join, as depicted to the left one in Figure 5.18 (with

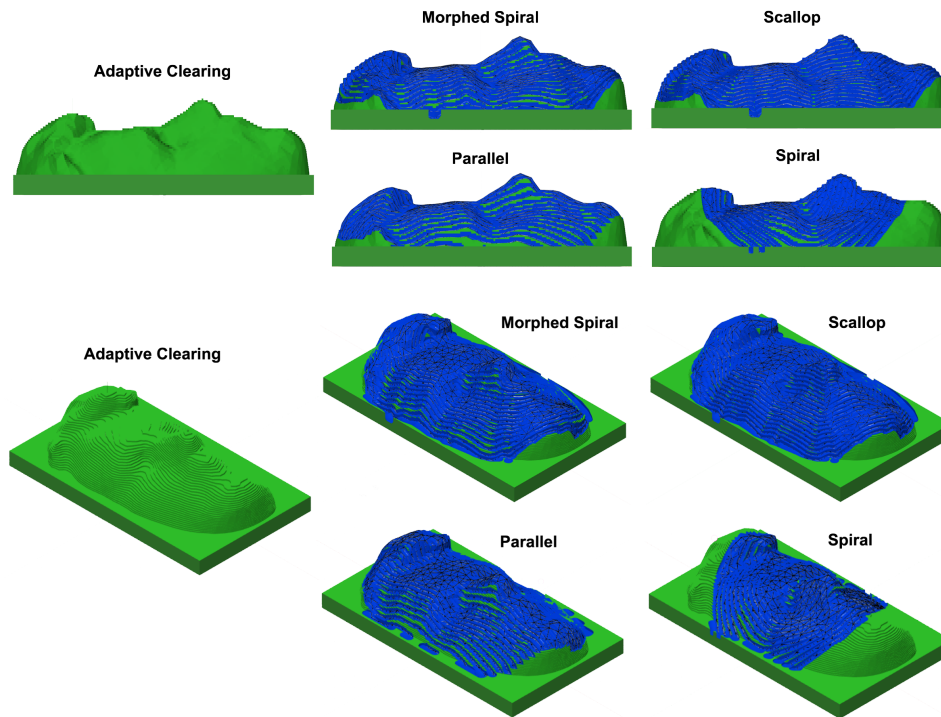


Figure 5.17: Simulation of 3D machining strategies in Fusion 360 with a profile and diagonal view.

three block of mesh merged), through the Merge function, making a combination of the meshes into a single smooth surface.

With the merged meshes, the Flow strategy is applied, depicted with a different colour of the Adaptive Clearing in the right image of Figure 5.18. This process is not possible to be applied to all meshes at once, besides having a high processing time to generate the toolpath with the Flow strategy.

In order to machine larger parts, due to tool restrictions, it was necessary to perform it in two stages, first machining the top of the part and then the bottom. Thus, two toolpaths were created with Adaptive Clearing, with differences in "Top Height" and "Bottom Height", as shown in Figure 5.19. Due to the dimensions of the milling cutter that has a free length of 30 mm, the part was generated with a height equal to twice the size of the tool and 1 mm more for the possible unevenness on surface, so having 61 mm of total height on the workpiece setup.

The respective Toolpath for each stage, together with the Fusion 360 simulation of both, can be viewed in Figure 5.20, being divided by colours of each operation.

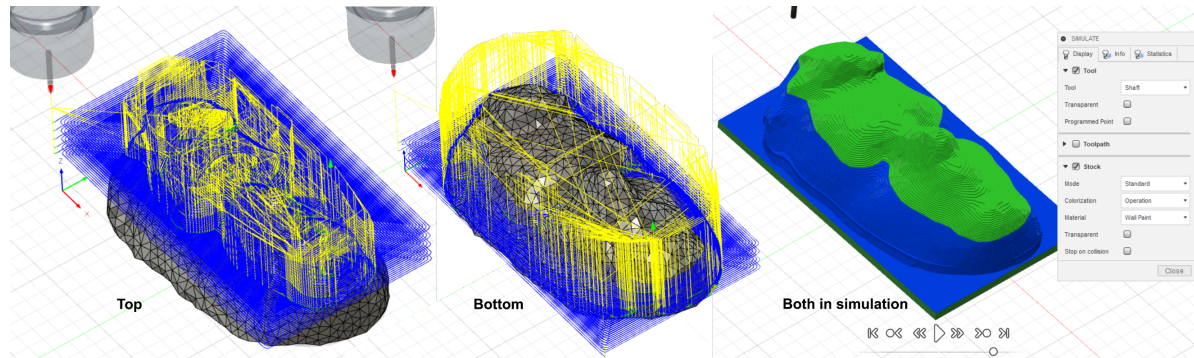


Figure 5.20: Top and bottom toolpath and respective simulation in Fusion 360.

5.3.3 RoboDK Simulations

The results of simulations in RoboDK were different from the other steps. They were used more to perform the toolpath tests of the previous step and to visualise machining positions to avoid errors during practice.

In addition to these tests, RoboDK made it possible to compare the IRB 2600 with the IRB 140 in machining tasks. It proved that in addition to the first being able to use a heavier tool due to its payload capacity, it also proved its machining performance in a larger area, which is of great importance for 3D machining, because in cases of machining the back of a part in relation to the robot, the tool must point towards the centre of the robot, thus reducing the working area. Figure 5.21 shows a test performed in simulation with the robot IRB 140, in which the toolpath generated could not be performed due to the robot not reaching some machining points.

5.3.4 Milling

In the practical milling process, acquisitions 1 and 4 of Figure 5.15 were used. The first was tested with a resized face 3 cm high from 3D acquisition 1, and the results were as shown in Figure 5.22.

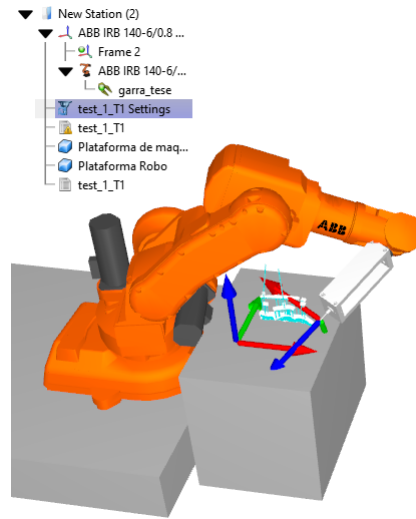


Figure 5.21: Problem of generating the robot code due to the limited reach of ABB IRB 140.

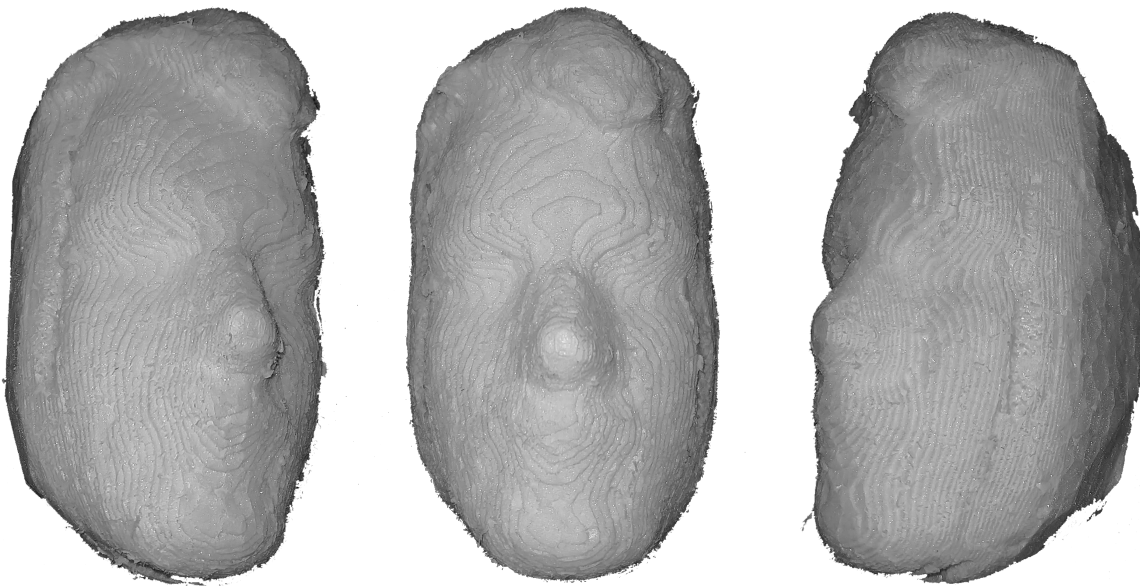


Figure 5.22: First milling with 30 mm height and without acceleration limitation.

In this first test, a thin layer was not roughed during the machining process, due to the stepover be considered equal to the cutter tool work length, and applied in a soft workpiece. Noting that if the characteristics of the cutter are the same as the roughing depth per layer, this will create a thin layer that is not roughened during the machining process. So in next tests, instead of using the working length of the milling cutter as the 6.2 mm of the real milling cutter, 6 mm was used to generate the toolpath; millings steps are smaller than the maximum height of the tool's working area. Also, the cutter was fixed on the support trying to get more roughing in the part, without collisions between the workpiece and the holder.

In the second test, the method of dividing the part into two layers was used, as shown in Figure 5.20, and thus made it possible to machine a larger part and better visualise the robot's toolpath, represented in Figure 5.23.

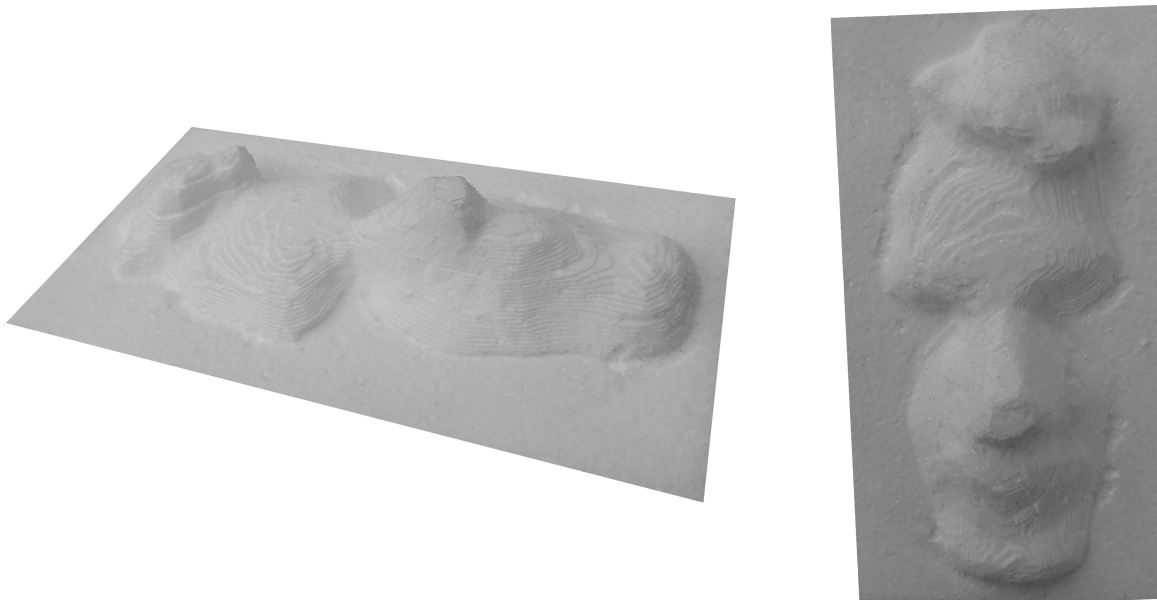


Figure 5.23: Top part milling with 60 mm height and without acceleration limitation.

With this step, it was possible to realise the importance of the work coordinate point, because by placing it on the top of the workpiece, it makes it easier to relate the real point to the point chosen in RoboDK. The real coordinate value is the same as the value in RoboDK when performing the top part operation, and for the bottom part operation it

must perform an offset on the z axis, due to its generated toolpath has the work coordinate reference point on top of the part, while the real acquired point is the machining begins coordinate.

In addition, the results showed major flaws due to the robot trembling when making sudden position change, so it was necessary to implement a limitation of maximum acceleration and deceleration of the robot (60 % in this work), to avoid these movements. The acceleration limitation was then used in all the following steps. First was reworked the top operation and after was performed the bottom part, showed in Figure 5.24.

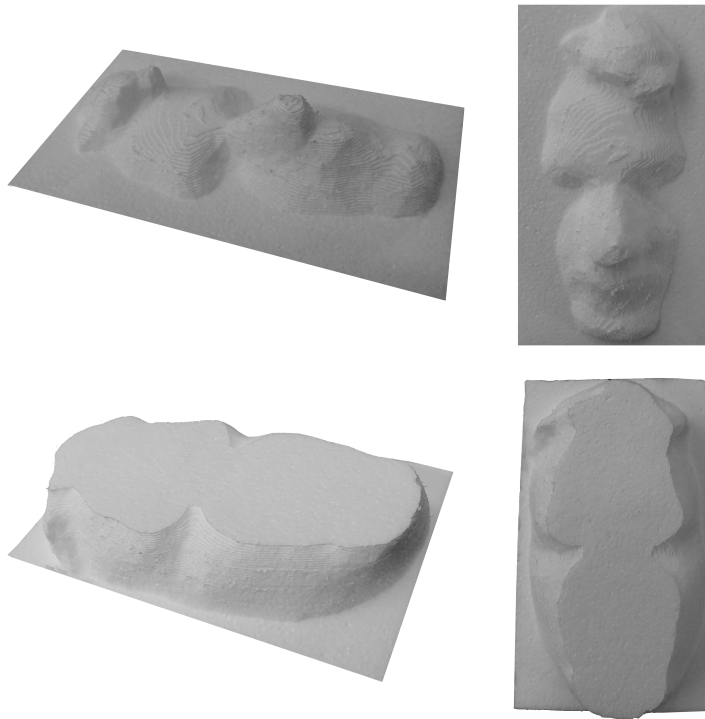


Figure 5.24: Milling with 60 mm height and with acceleration limitation.

The inclusion of the acceleration limitation has greatly reduced the failures due to the jerk caused without it. In the next test was performed the machining of a different face, shown in Figure 5.25, to see the performance.

With this same face, another test was performed using a different milling cutter (initially with the left one and in this step with the right one of Figure 4.8), in which only the top operation of this face was performed. This showed a visually different result, as

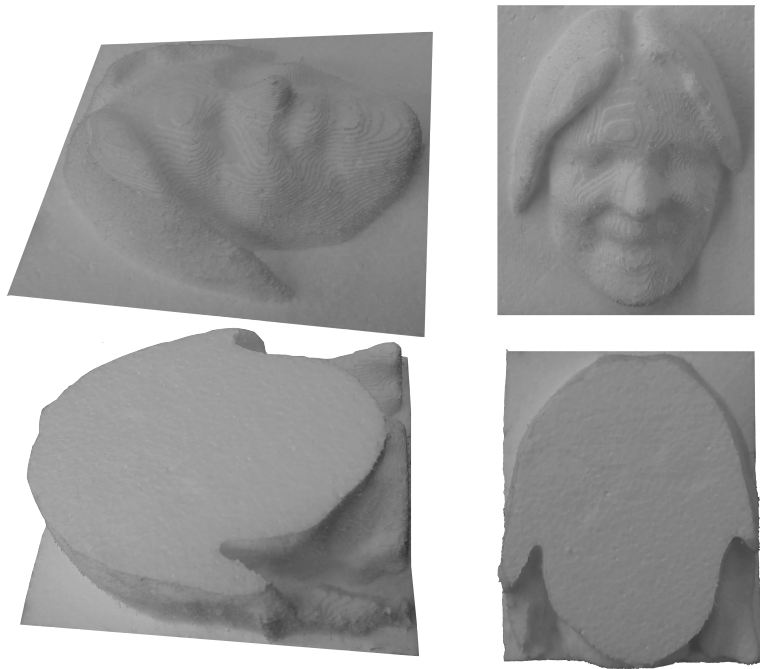


Figure 5.25: Milling with 60 mm height and with acceleration limitation of other face.

illustrated in Figure 5.26.

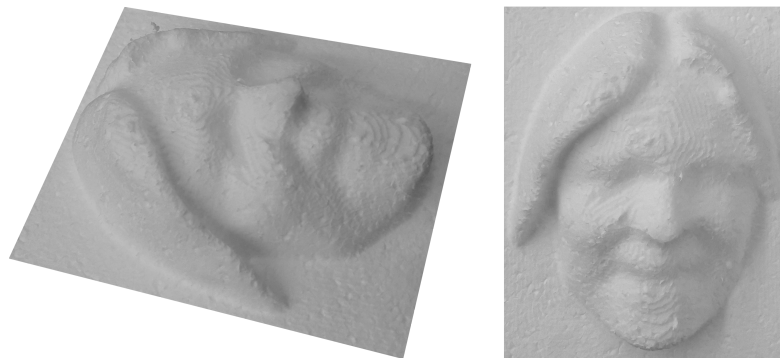


Figure 5.26: Milling the top part of the new face with other milling cutter.

Lastly the two top parts with the best results were cut to join the bottom part and compared to the corresponding machining simulation, as shown in Figure 5.27.

This image of the comparison of simulations and their respective real tests, confirmed that the results obtained by the processes met the expectations of the initial work objective, that is, using an industrial robot to perform the machining of a scene obtained by a

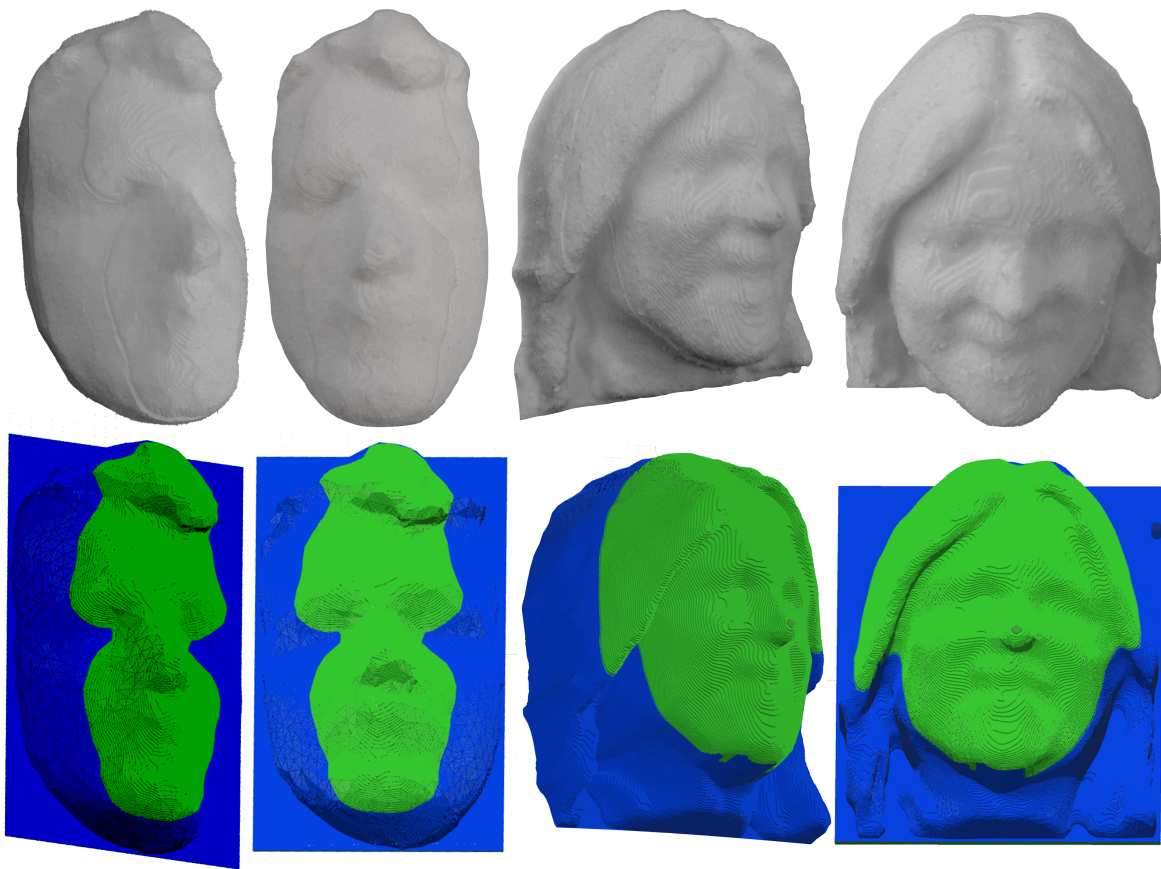


Figure 5.27: Faces in real (top images) and simulation milling(bottom images).

3D scanner. The visualisation of the whole precision between the simulation and the real part is not possible to be viewed in a figure, due to the colours and lighting available for simulation. However, this were better visualized in the software, due to the fact of being able to move the part to see it in other angles.

Because the milling processes took place with interruptions and switching from manual to automatic mode, the machining times were not recorded accurately. However it was estimated that the shortest time was 47 min for the part with 30 mm (Figure 5.22), and 145 min for the first top part with 60 mm (Figure 5.23). In addition, the times estimated by the simulation in RoboDK are described by Table 5.1.

Table 5.1: Estimated machining times on RoboDK.

Parts	30 mm height	Top 60 mm height	Bottom 60 mm height	Top new face	Bottom new face
Machining time	42 min	178 min	104 min	91 min	127 min

Chapter 6

Conclusion and Future Work

With this dissertation, a system for object reconstruction was developed, through 3D acquisition, acquisition adjustments, creation of the machining path and real machining of the object. Along with this, low-cost 3D acquisition options and the use of industrial robots in machining tasks were presented. In the work presented, human faces were used as models to perform the tests, with only two of the acquired faces and a roughing technique applied in practice.

3D image acquisitions were performed using the Intel Kinect with Skanect software, both to acquire the objects and for the faces. This acquisition generated faces with enough characteristics to differentiate each one, even without the small details, thus being suitable for the project objective and being a low cost choice for 3D image acquisitions. However due to the free export of the software, it reduced the characteristics of the faces used, and the tests proved that it affects more in parts with a lot of detail, which need an export with a larger number of meshes to not lose much of its details.

In addition, these acquisitions could be used in other applications, such as virtual storage of objects, company products and museum artefacts, or to remodel a given existing piece. However, the process of adjustments made in the acquisition of the work was only to remove the wholes, resize, cut the parts of interest and rotation.

Even though other machining techniques were used to finalise the part in the CAM software, these were not applied in practice because the available cutters did not generate

a big difference between the roughing and finishing process in simulation. Also the multi-axis machining was not as expected, being needed to use the Merge function, which is implemented manually and does not allow the Merge of the entire piece, but in parts. Furthermore, the time available to perform the tests was reduced, and chosen to perform other tests.

The RoboDK software was used to convert G-code into robot code, which was successful, in addition to providing a large number of robot models, and include a tool created from a CAD model that is able to update its TCP according to what was acquired in the real world. During the simulation in RoboDK was concluded that due to the cutters used, the roughing process took a long time and it limited the height of the part to be machined.

In the testes performed, it was possible to observe that an acceleration speed controller is required to reduce the vibration caused in the robot and in the workpiece, as both were not fixed to the floor.

At the end of the tests, it was observed that the stage that most lost of characteristic of the real model it was during the acquisition process, followed by the technique used for machining (due to the use of only the roughing process, without a finishing process). In the machining stage, the robot performed the machining process with a great accuracy (compared with the CAM simulation model), and even better with the addition of the acceleration control. Therefore, for a better result, another 3D acquisition method should have been implemented, use a multi axis finishing technique, and a larger cutter should be used in the roughing process.

In Conclusion, the work performed proved to be valid the use of industrial robots in machining services, these having a large work volume and being flexible, allowing to perform more tasks, such as positioning the part to be machined and remove it after machining.

6.1 Future Works

With the work presented, it was possible to identify points for its development that could be applied in future works. These possible developments are described in the following topics:

- Use different scanners for more accurate 3D acquisition even of small parts.
- Check the maximum mesh reduction without losing most of the part characteristics.
- Work with remodeling of existing parts, editing the 3d acquisition before machining.
- Fix the robot and the workpiece to the floor for reducing the vibrations.
- Use a larger cutter for roughing and a smaller one for the finishing process.
- Milling with different materials to better check the rigidity of the robot and to verify if the milling process has some disadvantages due to the Polystyrene flexibility.
- Test the multi-axis finishing machining.
- Try a reduction of software and automation of the processes used.

Bibliography

- [1] T. C. de Sousa Azevedo, “3d object reconstruction using computer vision: Reconstruction and characterization applications for external human anatomical structures”, PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2012.
- [2] P. C. Carvalho, L. Velho, A. Sá, E. Medeiros, A. A. Montengro, A. Peixoto, and L. A. R. Escriba, *Fotografia 3D*. Rio de Janeiro: Associação Instituto de Matemática Pura e Aplicada, IMPA, 2005.
- [3] Y.-L. Liu and T.-H. Lin, “Feature enhancement for a defocusing structured-light 3-d scanning system”, *Optical Engineering*, vol. 57, no. 6, pp. 1–11, 2018. DOI: 10.1117/1.OE.57.6.064101.
- [4] M. Cheng, *Visualisation of superficial breast changes*, 1998.
- [5] S. S. Moreira, “Projeto e construção de um scanner tridimensional baseado no método de triangulação a laser utilizando softwares livres”, Universidade Estadual Paulista (UNESP), 2016.
- [6] M. E. Polo, A. Cuartero, and Á. M. Felicísimo, “Study of uncertainty and repeatability in structured-light 3d scanners”, *arXiv e-prints*, 2019. DOI: arXiv:1910.13199.
- [7] E. P. d. Santos, “Cad/cam/usinagem cnc integrado a engenharia reversa”, Universidade Estadual Paulista (UNESP), 2017.
- [8] P. M. F. Almeida, “Programação de robôs para realização de tarefas de maquinação”, Master’s thesis, Instituto Superior de Engenharia do Porto, 2016.

- [9] J. Wallén, *The history of the industrial robot*. Linköping University Electronic Press, 2008.
- [10] A. Gasparetto and L. Scalera, “A brief history of industrial robotics in the 20th century”, *Advances in Historical Studies*, vol. 8, no. 1, pp. 24–35, 2019.
- [11] J. Angeles, *Fundamentals of robotic mechanical systems*. Springer, 2002.
- [12] M. Mihelj, T. Bajd, A. Ude, J. Lenarčič, A. Stanovnik, M. Munih, J. Rejc, and S. Šlajpah, *Robotics*. Springer, 2019.
- [13] D. F. C. Angelino, “Conceção e simulação de uma célula robótica para operações de acabamento de guitarras feitas em materiais compósitos”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2012.
- [14] D. J. M. Malheiro, “Dispositivo automático para operações de acabamento”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2014.
- [15] W. Ji and L. Wang, “Industrial robotic machining: A review”, *The International Journal of Advanced manufacturing Technology*, vol. 103, no. 1-4, pp. 1239–1255, 2019. DOI: 10.1007/s00170-019-03403-z.
- [16] J. P. M. Barbosa, “Concepção e simulação de célula robotizada para operações de acabamento”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2010.
- [17] S. Mejri, V. Gagnol, T.-P. Le, L. Sabourin, P. Ray, and P. Paultre, “Dynamic characterization of machining robot and stability analysis”, *International Journal of Advanced Manufacturing Technology*, Jun. 2015. DOI: 10.1007/s00170-015-7336-3.
- [18] S. Mousavi, V. Gagnol, B. Bouzgarrou, and P. Ray, “Model-based stability prediction of a machining robot”, in *New advances in mechanisms, mechanical transmissions and robotics*, Springer, 2017, pp. 379–387.

- [19] J. Gutiérrez, H. Chanal, S. Durieux, and E. Duc, “Adaptation of the geometric model of a 6 dof serial robot to the task space”, in *Computational Kinematics*, Springer, 2018, pp. 569–576.
- [20] C. Lehmann, M. Halbauer, D. Euhus, and D. Overbeck, “Milling with industrial robots: Strategies to reduce and compensate process force induced accuracy influences”, in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, IEEE, 2012, pp. 1–4.
- [21] L. M. F. Miranda, “Maquinagem robotizada para customização de produtos”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2020.
- [22] M. Pollák, M. Telišková, M. Kočiško, and P. Baron, “Application of industrial robot in 5-axis milling process”, in *MATEC Web of Conferences*, EDP Sciences, vol. 299, 2019, pp. 1–6.
- [23] R. Rust, D. Jenny, F. Gramazio, and M. Kohler, “Spatial wire cutting: Cooperative robotic cutting of non-ruled surface geometries for bespoke building components”, 2016.
- [24] F. Alhwarin, A. Ferrein, A. Gebhardt, S. Kallweit, I. Scholl, and O. Tedjasukmana, “Improving additive manufacturing by image processing and robotic milling”, in *International Conference on Automation Science and Engineering (CASE)*, IEEE, 2015, pp. 924–929.
- [25] C. Wu, C. Dai, G. Fang, Y.-J. Liu, and C. C. Wang, “Robofdm: A robotic system for support-free fabrication using fdm”, in *International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 1175–1180.
- [26] J. Pandremenos, C. Doukas, P. Stavropoulos, and G. Chryssolouris, “Machining with robots: A critical review”, *Proceedings of DET2011*, pp. 1–9, 2011.
- [27] Y. Hu and Y. Chen, “Implementation of a robot system for sculptured surface cutting. part 1. rough machining”, *The International Journal of Advanced Manufacturing Technology*, vol. 15, no. 9, pp. 624–629, 1999.

- [28] ———, “Implementation of a robot system for sculptured surface cutting. part 2. finish machining”, *The International Journal of Advanced Manufacturing Technology*, vol. 15, no. 9, pp. 630–639, 1999.
- [29] W. S. Owen, E. A. Croft, and B. Benhabib, “Real-time trajectory resolution for dual robot machining”, in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, IEEE, vol. 5, 2004, pp. 4332–4337.
- [30] S. E. Sarma, P. Laxmiprasad, and M. Balasubramaniam, *Tool path generator for computer aided manufacturing*, US Patent 6,311,100, Oct. 2001.
- [31] Y. Chen and F. Dong, “Robot machining: Recent development and future research issues”, *The International Journal of Advanced Manufacturing Technology*, vol. 66, no. 9-12, pp. 1489–1497, 2013.
- [32] W. H. Walstra, W. F. Bronsvort, and J. S. Vergeest, “Interactive simulation of robot milling for rapid shape prototyping”, *Computers & graphics*, vol. 18, no. 6, pp. 861–871, 1994.
- [33] V. M. Oliveira, “Uso de arquitetura embarcada para automação do processo de aquisição de imagens para escaneamento 3d de baixo custo”, Master’s thesis, Engenharia Eletrônica da Universidade de Brasília, 2018.
- [34] J. P. A. Duarte, “Desenvolvimento de uma plataforma de impressora 3d acoplada a um scanner 3d”, 2017.
- [35] R. J. C. de Miranda, “Desenvolvimento de um programa didático computacional destinado à geração de códigos de comando numérico a partir de modelos 3d obtidos em plataforma cad considerando a técnica prototipagem rápida”, 2009.
- [36] L. C. P. L. Oliveira, “Maquinagem de superfícies complexas com recurso a sistema robótico”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2013.

- [37] M. Pollák, J. Török, J. Zajac, M. Kočiško, and M. Telišková, “The structural design of 3d print head and execution of printing via the robotic arm abb irb 140”, in *2018 5th International Conference on Industrial Engineering and Applications (ICIEA)*, IEEE, 2018, pp. 194–198.
- [38] A. B. Prasai, J. Greebmalai, and E. Warinsiriruk, “Analysis of industrial robot arm programming for 3d part machining”, *The Conference of Industrial Engineering Network*, 2020.
- [39] M. Karczewski, “Influence of 3d scanner parameters on accuracy evaluation of vehicle element deformation”, in *AIP Conference Proceedings*, AIP Publishing LLC, vol. 2078, 2019, p. 020012.
- [40] —, “Zastosowanie metod inżynierii odwrotnej do identyfikacji obiektów technicznych”, *TTS Technika Transportu Szybowego*, vol. 22, 2015.
- [41] D. Lanman and G. Taubin, “Build your own 3d scanner: 3d photography for beginners”, in *ACM SIGGRAPH 2009 Courses*, 2009, pp. 1–94.
- [42] F. Bernardini and H. Rushmeier, “The 3d model acquisition pipeline”, in *Computer graphics forum*, Wiley Online Library, vol. 21, 2002, pp. 149–172.
- [43] R. Sitnik, M. Karaszewski, W. Załuski, and J. Rutkiewicz, “Automation of 3d scan data capturing and processing”, in *Three-Dimensional Image Processing (3DIP) and Applications*, International Society for Optics and Photonics, vol. 7526, 2010, 75260Y.
- [44] B. Curless and S. Seitz, “3d photography”, *Course Notes for SIGGRAPH 2000*, 2000.
- [45] S. Seitz, “An overview of passive vision techniques”, *SIGGRAPH 2000 Course on 3D Photography, Course Notes, New Orleans, Louisiana, LA, USA*, 1999.
- [46] P. C. P. Júnior, A. Von Wangenheim, A. C. Sobieranski, and S. L. M. Neto, “Desenvolvimento de um scanner 3d sem contato de baixo custo usando marcadores artificiais”, *Anais do Computer on the Beach*, pp. 310–319, 2016.

- [47] A. Gruen, F. Remondino, and L. Zhang, “Image-based automated reconstruction of the great buddha of bamiyan, afghanistan”, in *2003 Conference on Computer Vision and Pattern Recognition Workshop*, IEEE, vol. 1, 2003, pp. 13–13.
- [48] M. Gazziro, *Scanner 3D-Construção e Aplicações*. Biblioteca24horas, 2011.
- [49] P. Suliga, “A feature analysis of a laser triangulation stand used to acquire a 3d screw thread image”, in *2016 17th International Carpathian Control Conference (ICCC)*, IEEE, 2016, pp. 702–705.
- [50] J. Geng, “Structured-light 3d surface imaging: A tutorial”, *Advances in Optics and Photonics*, vol. 3, no. 2, pp. 128–160, 2011.
- [51] Y. Zhang, “A structured light based 3d reconstruction using combined circular phase shifting patterns”, PhD thesis, The Ohio State University, 2019.
- [52] I. Anwar and S. Lee, “High performance stand-alone structured light 3d camera for smart manipulators”, in *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, IEEE, 2017, pp. 192–195.
- [53] C. D. Mutto, P. Zanuttigh, and G. M. Cortelazzo, *Time-of-flight cameras and microsoft kinect (TM)*. Springer Publishing Company, Incorporated, 2012.
- [54] L. C. T. Coelho Filho, J. L. Nunes, and S. Brito, “Projeto e-foto: Uma estação fotogramétrica digital educacional”, *Undergraduate project, Brazilian Military Institute of Engineering*, 2002.
- [55] Z. Majid, A. K. Chong, A. Ahmad, H. Setan, and A. R. Samsudin, “Photogrammetry and 3d laser scanning as spatial data capture techniques for a national craniofacial database”, *The Photogrammetric Record*, vol. 20, no. 109, pp. 48–68, 2005.
- [56] A. M. G. Tommaselli, “Um estudo sobre as técnicas de varredura a laser e fotogrametria para levantamentos 3d a curta distância”, *GEODÉSIA Online-Revista da Comissão Brasileira de Geodésia*, vol. 4, no. 1415-1111, 2003.

- [57] R. Karnicki, “Photogrammetric reconstruction software as a cost-efficient support tool in conservation research”, *Technical Transactions*, vol. 117, no. 1, 2020.
- [58] N. S. Potabatti, “Photogrammetry for 3d reconstruction in solidworks and its applications in industry”, PhD thesis, Purdue University College of Engineering, 2019.
- [59] AliceVision, *Photogrammetric computer vision framework*, <https://alicevision.org/>, last accessed on 05/09/20.
- [60] D. Catuhe, *Programming with the Kinect for Windows software development kit*. Pearson Education, 2012, ISBN: 978-0-7356-6681-8.
- [61] G. S. Cardoso, *Microsoft Kinect: Crie aplicações interativas*. Casa do Código, 2014, ISBN: 978-85-66250-21-3.
- [62] S. Liu, W. Li, P. Ogunbona, and Y.-W. Chow, “Creating simplified 3d models with high quality textures”, in *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, 2015.
- [63] O. Manikhi and B. Adlkhast, *A 3d object scanner: An approach using microsoft kinect*, 2013.
- [64] D. d. S. A. Loura, Y. d. M. A. de Oliveira, P. O. Raimundo, and K. P. A. Agüero, “Reconstrução 3d de objetos com kinect e câmera digital”, *Revista Eletrônica de Iniciação Científica em Computação*, vol. 16, no. 6, 2018.
- [65] W. R. Abdulmajeed and R. Z. Mansoor, “Implementing kinect sensor for building 3d maps of indoor environments”, *International Journal of Computer Applications*, vol. 86, no. 8, 2014.
- [66] G. Gatto, “Escaneo 3d y estudioaerodinámico de un automóvil de competición”, 2016.

- [67] Intel, *Depth quality testing white paper for intel realsense camera*, <https://www.intel.com/content/www/us/en/support/articles/000026982/emerging-technologies/intel-realsense-technology.html>, last accessed on 13/09/20, Jan. 2018.
- [68] M. Omori, “The development of a body composition imaging system using 3d depth sensors utilizing machine learning for testing in microgravity simulations”, PhD thesis, University of Hawai’i at Manoa, 2020.
- [69] RecFusion, *Turn your world into a 3d model*, <https://www.imfusion.com/products/recfusion>, last accessed on 14/09/20.
- [70] LIPScan, *3d scan for professionals and hobbyists*, <https://www.lips-hci.com/lipscan>, last accessed on 14/09/20.
- [71] R. B. Rusu and S. Cousins, “3d is here: Point cloud library (pcl)”, in *2011 IEEE international conference on robotics and automation*, IEEE, 2011.
- [72] O. Wulf and B. Wagner, “Fast 3d scanning methods for laser measurement systems”, in *International conference on control systems and computer science*, Cite-seer, 2003, pp. 2–5.
- [73] P. Dorninger and N. Pfeifer, “A comprehensive automated 3d approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds”, *Sensors*, vol. 8, no. 11, pp. 7323–7343, 2008.
- [74] G. Vosselman, B. G. Gorte, G. Sithole, and T. Rabbani, “Recognising structure in laser scanner point clouds”, *International archives of photogrammetry, remote sensing and spatial information sciences*, vol. 46, no. 8, pp. 33–38, 2004.
- [75] T. Brito, “Intelligent collision avoidance system for industrial manipulators”, Master’s thesis, Polytechnic Institute of Bragança, 2019.
- [76] S.-H. Huang, L.-C. Zhang, and M. Han, “An effective error-tolerance slicing algorithm for stl files”, *The International Journal of Advanced Manufacturing Technology*, vol. 20, no. 5, pp. 363–367, 2002.

- [77] W. Li and P. Song, “A modified icp algorithm based on dynamic adjustment factor for registration of point cloud and cad model”, *Pattern Recognition Letters*, vol. 65, pp. 88–94, 2015.
- [78] CONSELF, *From point cloud to stl triangulated surface*, <https://consself.com/blog/from-point-cloud-to-stl-triangulated-surface/>, last accessed on 15/09/20.
- [79] Autodesk Meshmixer, *Free software for making awesome stuff*, <https://www.meshmixer.com/>, last accessed on 15/09/20.
- [80] G. Chryssolouris, *Manufacturing systems: theory and practice*. Springer Science & Business Media, 2013.
- [81] I. R. Sharma, “Latest trends in machining”, *Retrieved July 19, 2011, from Indra Drishtikona: http://www.drishtikona.com/books/latest-trends-in-machining/ch-all.pdf*, 2001.
- [82] J. Pandremenos, C. Doukas, P. Stavropoulos, and G. Chryssolouris, “Machining with robots: A critical review”, *Proceedings of DET2011*, pp. 1–9, 2011.
- [83] L. C. P. L. Oliveira, “Percursos de maquinagem 2d: Mesa cnc e robô industrial”, Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 2018.
- [84] H. Zhang, J. Wang, G. Zhang, Z. Gan, Z. Pan, H. Cui, and Z. Zhu, “Machining with flexible manipulator: Toward improving robotic machining performance”, in *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, IEEE, 2005, pp. 1127–1132.
- [85] IFR, “Executive summary world robotics 2017 industrial robots”, 2017.
- [86] Autodesk, Inc., *Fundamentals of CNC Machining: A Practical Guide for Beginners*. Autodesk, 2014.
- [87] ISO Central Secretary, “Robots and robotic devices — vocabulary”, en, International Organization for Standardization, Standard ISO 8373:2012, 2012, <https://www.iso.org/standard/55890.html>, last accessed on 15/09/20.

- [88] Y. Koren and Y. Koren, *Robotics for engineers*. McGraw-Hill New York, 1985, vol. 168.
- [89] M. P. Groover, *Robótica: tecnologia e programação*. McGraw-Hill, 1989.
- [90] Vision spectra, *The integration of machine vision with motion control is transforming bin picking, part inspection*, https://www.photonics.com/Articles/The_Integration_of_Machine_Vision_with_Motion/a64353, last accessed on 16/09/20.
- [91] S. Y. Nof, *Handbook of industrial robotics*. John Wiley & Sons, 1999.
- [92] C. E. Fanti, W. R. B. Ferreira, and J. C. M. Carvalho, “Planejamento de trajetórias robóticas definidas por segmentos de reta concordantes por polinomiais”, *Horizonte Científico*, 2011.
- [93] P. Michalik, J. Zajac, J. Duplák, and A. Pivovarnik, “Cam software products for creation of programs for cnc machining”, in *Future Communication, Computing, Control and Management*, Springer, 2012, pp. 421–425.
- [94] M. P. Dimitrov and S. I. Antonov, “Application of artificial intelligence and machine learning in cad/cam systems”, unpublished, 2019.
- [95] *SolidWorks Corp.* <https://www.solidworks.com/product/solidworks-cam>, last accessed on 21/09/20.
- [96] P. Song, Y. Qi, and D. Cai, “Research and application of autodesk fusion360 in industrial design”, in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, vol. 359, 2018, p. 012 037.
- [97] Autodesk, Inc., *Integrated cad, cam, cae, and pcb software*, <https://www.autodesk.com/products/fusion-360/overview>, last accessed on 22/09/20.
- [98] Mastercam, *Mastercam creates software and services that solve the world’s manufacturing challenges*, <https://www.mastercam.com/>, last accessed on 22/09/20.
- [99] CNC Cookbook, *Cam software market share*, <https://www.cnccookbook.com/results-2017-cnccookbook-cam-survey/>, last accessed on 22/09/20.

- [100] W.-j. Shan, “Parameter setting for dynamic milling of aluminum alloy”, in *2017 International Conference on Manufacturing Engineering and Intelligent Materials (ICMEIM 2017)*, Atlantis Press, 2017.
- [101] A. C. Brown and D. De Beer, “Development of a stereolithography (stl) slicing and g-code generation algorithm for an entry level 3-d printer”, in *2013 Africon*, IEEE, 2013, pp. 1–5.
- [102] M. P. S. Dantas and O. V. D. S. Junior, “Fresadora cnc de baixo custo - susy”, *Mostra Nacional de Robótica (MNR)*,
- [103] A. S. Jurisato, M. R. Rocha, L. A. V. Hellmeister, J. F. Marar, and A. C. Sementille, “Difusão de tecnologia cad e cnc como ferramenta básica de criação de produção em pequena escala, acessível à comunidade”, in *Congresso de Extensão Universitária*, Universidade Estadual Paulista (UNESP), 2013, p. 09 892.
- [104] RoboDK, *Program any industrial robot with one simulation environment*, <https://robodk.com/>, last accessed on 24/09/20.
- [105] ABB, *Irb 2600: Abb further extends its mid range robot family*, <https://new.abb.com/products/robotics/industrial-robots/irb-2600>, last accessed on 02/10/20.
- [106] —, *IRC5: Industrial robot controller*, <https://new.abb.com/products/robotics/controllers/irc5>, last accessed on 02/10/20.
- [107] DREMEL, *DREMEL 3000/4000/4300*, https://www.dremeleurope.com/za/en/ocsmedia/2610Z09984_InstrMan_3000_4000_4300_EU_196_JUL18_LR.pdf, last accessed on 03/10/20.
- [108] —, *ACESSÓRIOS*, <https://www.dremeleurope.com/pt/pt/products/accessories/?start=0&System=Sistema+de+multiferramentas>, last accessed on 04/10/20.