Theoretical approach of computer-aided shape design and manufacturing simulation to dental wire bending

Rahimah binti Abdul Hamid

ACKNOWLEDGEMENTS

In the name of Allah, the most Merciful, the most Gracious. All praise is due to Allah; we praise Him, seek His help, and ask for His forgiveness. I am thankful to Allah, who supplied me with the courage, the guidance, and the love to complete this research. I cannot forget the ideal man of the world and most respectable personality for whom Allah created the whole universe, Prophet Muhammad S.A.W.

Undertaking this Ph.D. has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people. Because this may be my only opportunity to thank these individuals in writing, I may be a bit more verbose in my thanks than necessary. I gratefully acknowledge the funding received towards my Ph.D. from the Universiti Teknikal Malaysia Melaka (UTeM) Ph.D. scholarship and the Ministry of Higher Education, Malaysia.

As my Ph.D. supervisor, Professor Dr. Teruaki Ito deserves thanks for many things. Notably, for creating the research environment in which I have performed my graduate studies. I came to Tokushima University intending to join Collaborative Engineering Laboratory (CELab), for two primary reasons: the areas of research in which he was involved, and what I had heard about his personality. I have not been disappointed in either regard. He has provided guidance at key moments in my work while also allowing me to work independently the majority of the time.

Because of the research environment sustained by CELab, I have crossed paths with many undergraduate students and graduate students from who have influenced and enhanced my research. The direction of my graduate work has been strongly influenced by the members of the CELab (2015-2018), notably Kiritate Kosuke, Kawakami Yuki, Jyo You, Norazlin Nasir, Teoh Yong Siang, Madihah Maharof, Morisaki Kenta, Kajioka Fumitaka, Ohkita Hiroki, Satoru Iwami, Toshitaka Higashikoba, Akihiko Hanazaki, Kimachi Hiroki, Wang Yakai, Afiqah Ngah Nasaruddin and Wan Norhafiz Wan Ahmad.

I also owe a debt of gratitude to all the members of the doctoral committee for their unfailing academic helpfulness throughout this research study, namely Professor Masaki Hashizume, Professor Mitsuhiro Ohta, Professor Tetsuo Ichikawa, and Professor Yoshihiro Deguchi. The influence of those with whom I worked before coming to Tokushima University also continues to be important. Professor Dr. Vijay R. Raghavan provided me the foundations that allowed me to hit the ground running at Tokushima

University and complete my thesis in three years. Prof. Vijay helped my writing skills to flourish. Thanks to the Vice Chancellor, Deputy Vice Chancellor (Academic & International), and the Dean of the Faculty of Manufacturing Engineering, Universiti Teknikal Malaysia Melaka (UTeM), Prof. Datuk Ts. Dr. Shahrin bin Sahib@Sahibuddin, Professor Datuk Dr. Mohd. Razali bin Muhamad and Associate Professor Dr. Rizal bin Salleh for approving my study leave to Tokushima University.

I also extend my heartfelt appreciation to my family; I have been very fortunate to have their encouragement, confidence, and love for many years. My parents and my parents-in law, whose selfless sacrificial life and their great efforts with pain and tears and unceasing prayers have enabled me to reach the present position in life. I should not forget to acknowledge my brothers and sisters for their continuous love, support, understanding, and good wishes whenever I needed.

Special thanks to my husband, Mohd. Yushairee bin Md. Yusoff whose great patience, love and strong emotional support made my life pleasant even in the hardest times. You have been continually supportive of my graduate education. Thank you for the many things you have done. Thank you for taking care of the kids when I went to the laboratory, even if I had to stay late at work on weekdays or weekends. You have been patient with me when I am frustrated, you celebrate with me when even the littlest things go right, and you are there whenever I need you to just listen. I love you more than you will ever know. Thanks to my kids; Irfan Danish, Razin Danish and Alyna Danish for all your patience and support during mommy's Ph.D. studies. I love you!

Also thanks to Ms. Michiko Shinohara from the Student Affairs Section, Graduate School of Advanced Technology and Science, Tokushima University for always helping me with my life in Japan. Thanks to Ms. Norsuria binti Azis from the Study Leave Division, Universiti Teknikal Malaysia Melaka (UTeM) for always answering my doubts and taking care of my financial aid throughout my Ph.D. study.

Last but not least, I want to express my deepest gratitude to my friends Nurdiana binti Nordin, Siti Rahmah binti Shamsuri, Fara Ashikin binti Ali, Masni-Azian binti Akiah, Toibah binti Abd. Rahim and Mohd. Shahrizan bin Othman for their love, understanding and their support. Finally, I thank all those who have helped me directly or indirectly in the successful completion of my thesis. Anyone missed in this acknowledgment is also thanked.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABAM	Automated Bending Arch-wire Machine
BAS	Bending Art System
B-code	Bending code
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CNC	Computer-Numerical Control
IGES	Initial Graphics Exchange Specification
LAMDA	Lingual Arch-wire Manufacturing and Design Aid
NiTi	Nickel Titanium
STL	Stereolithography
SLA	Stereolithography
TMA	Titanium Molybdenum

LIST OF SYMBOLS

Symbols	Description	Unit
L	Length	mm
β	Rotation angle	Degrees
θ	Bending angle	Degrees
α	Included angle	Degrees
r	Bend radius	mm
K_S	Spring back factor	-

ABSTRACT

Dental wire bending in orthodontics and prosthodontics applications are traditionally performed in the hand-made operation. Recent advancements in the field of arch-wire bending point to several attempts to automate the dentistry wire bending operation through Computer-Numerical Control (CNC) dental wire bending machine and robotic device. Automated dental wire bending system helps to eliminate all the drawbacks associated with manual wire bending operation, and at the same time offers some advantages over the traditional approach. The review related to dental wire bending systems for the past two decades illustrates that all of the reported inventions have focused on arch-wire bending for both labial and lingual orthodontic. This highlights the demand for a more flexible dental wire bending system which could be used in other wire-related dentistry treatments, not just to bend a flat arch-wire or the customized arch-wire. For this reason, the present study tries to propose the concept of a flexible dental wire bending system by choosing a CNC dental wire bending machine to execute the desired 3D wire bending tasks, as the first objective of the study.

In order to get started, a theory of 3D linear segmentation algorithm is introduced. Digital wire design in dentistry is complicated due to freeform shapes which make them difficult to be predefined generically. As a result, the conversion process from Computer-Aided Design (CAD) to Computer-Aided Manufacturing (CAM) data would be impossible. Therefore, in this theory, the freeform shape is segmented into multiple 3D lines, where the XYZ Cartesian coordinates of the start and the end point in every line segment are used for the generation of wire bending parameters, known as the bending code (B-code). In theory, the bending of a 3D real curve is made based on the relative spatial parameters (L-the length, β-the rotation angle, and θ- the bend angle,) of the adjacent 3D line segments. B-code controls the bending operation of the CNC machine in this way; L tells the feeding mechanism the desired bend length to be fed, β instructs the turning mechanism to rotate if there is a change of plane in the subsequent bending while θ controls the bending mechanism to create the desired bend angle in each bending operation.

For the purpose of understanding the theory of B-code shape definition, a 3D simulation using a sheet metal feature is carried out by reverse engineering approach. The actual pre-bent labial bow is re-constructed in this feature and the simulated relative spatial parameters are compared with the calculated B-code parameters for validation. As a result, the mathematical formulae proposed in the theory of 3D linear segmentation are relevant for the subsequent development of the B-code generation program. In addition, the manufacturing workflow to process the initial digital teeth data towards the automated

3D wire bending is proposed which consists of 3D data digitizing, digital shape generation, XYZ coordinates extraction, B-code generation, and 3D dental wire bending. In the first stage, different types of 3D digitizing methods have been considered which led to different types of input data for the subsequent digital shape generation process. This refers to the digital wire design in the computer, in relation to the planning of bending points' algorithm. Therefore, some interfaces to process the variation of input files towards the XYZ coordinates extraction have been discussed, in terms of the stereolithography (STL) file, the 3D point cloud data and the 2D image file. Also, an automatic XYZ coordinate's extraction program which is capable to extract the XYZ coordinates from the Initial Graphics Exchange Specification (IGES) file has also been established. In the fourth stage of the manufacturing workflow, an automatic B-code generation program has been developed to automatically produce the B-code data from the XYZ coordinates data. This reflects the second objective of the study.

Additionally, the implementation of the B-code is demonstrated through a case study which simultaneously tested the IGES-XYZ coordinate extraction program and the B-code generation program. Furthermore, 3D wire bending simulations of the generated B-code which are initially processed by the 3D point cloud data filtering and the 3D point cloud data selection programs are executed in order to check the reliability of the introduced programs. Consecutively, the generated B-code from the 2D image input file has also been examined. These have proven the ability of the B-code generation program to provide the required B-code regardless of the type of input data. Finally, the feasibility of the generated B-code towards realizing a successful 3D wire bending operation has been validated, as the third objective of the study.

Other than that, a mathematical definition of human dental arch form has also been additionally investigated, in an attempt to design an optimized arch shape for the braces treatment. This could improve the theory of the 3D linear segmentation into a more semi-optimized curve. However, the actual arch-wire bending operation to realize the semi-optimized arch shape has not been studied, which remains as the recommendation for future work. To conclude, the research objectives in this study have been achieved, focusing on the introduction of bending points' algorithm and the theoretical manufacturing of personalized target shape. As for future work, the interface to directly integrate the B-code generation program to the CNC wire bender has to be developed, so that they could work consecutively to perform the desired wire bending operation. Also, the planning of wire spring back algorithm towards the theoretical bending angle (θ) could also be scrutinized.

CHAPTER 1

INTRODUCTION

1.0 Background

The present study proposes a concept to automate the dentistry wire bending operation by means of Computer-Numerical Control (CNC) machine. Till recently, a few advancements in the field of arch-wire bending for orthodontics treatment have been reported by researchers. Orthodontics is the branch of dentistry concerned with the prevention, interception, and correction of malocclusions and dental displacements through the wearing of braces [1]. These inventions, however, are only focusing on the arch-wire bending whereas there are many other wire-related treatments in prosthodontics and orthodontics applications, like I-bar clasp and C-clasp which are still traditionally produced. In addition, robotic devices are mostly employed to realize the wire bending operation in comparison to the CNC wire bending machine. Furthermore, the application of the introduced systems is only limited to a customized arch-wire and a flat arch-wire manufacturing for the orthodontic treatment. Apparently, robotic technology and orthodontics application have generated more interest among the researchers for an unknown reason, despite the complexity of the robotic system, in addition to the high cost of the robot. Consequently, a portable CNC dental wire bending machine which is more easy to use and less expensive is more relevant to be used during the chair-side treatment. A review of the developed CNC dental wire bending machine indicates that the introduced systems are only capable to bend a flat arch-wire and incapable to produce any complex and twist bend in the wire, which requires a combination of translational and rotational motion, contrary to the robotic devices which could realize a 3D wire bending and produce a customized arch-wire. Taking this factor as a consideration, a dental wire bending machine which is more flexible and could be used to bend any type of inputs given would be beneficial to the industry. Moreover, the use of CNC wire bending technology towards dentistry application is still not widely explored since only a few inventions related to dental wire bending machine has been reported, including those which are still at theoretical research and preliminary experiment level. Therefore, the present study

explores the possibility of using the CNC wire bending technology and attempts to establish a groundwork for realizing the concept. CNC wire bending machine is chosen over the robotic device due to the complexity and high cost of the robot. This could offer the use of modern technology in assisting the wire bending operation on a broader scale for orthodontics and prosthodontics applications in the future.

The use of Computer-Aided Design (CAD) and Computer Aided Manufacturing (CAM) in dental restorations has become popular over the past 30 years [2]. CAD/CAM technology for dental restorations [3] has numerous advantages over traditional techniques. Dentistry wire bending automated systems, on the other hand, has started to draw attention among researchers. The traditional approach of dentistry wire bending in orthodontics and prosthodontics applications requires a manual operation which is based on clinical experience [4] and has high tendency to create errors [5]. Cooperation between experienced dental specialist and a skilled dental technician is often needed to produce high-quality dentures with low revision rate, due to the shape complexity and the uncertainty of manual operation [6]. This has caused a high level of randomness and put many restrictions on clinical prosthodontics and orthodontics. Alternatively, the automated wire bending system eliminates all the drawbacks in manual wire bending operation and reduce the treatment time [7], in comparison to the conventional approach. In addition, dental CAD/CAM system provides the availability of virtual cyber-dental space, where the treatment information could be stored virtually and shared among the dental expert and highly skilled technician [8]. As a consequences, there is a great demand for an automated wire bending system which is portable, easy to operate and could realize a more flexible wire bending to compensate all of the limitations associated with manual wire bending operation.

1.1 Problem statement

The reported inventions in the field of dentistry wire bending are all focusing on the arch-wire bending for braces treatment in orthodontics. Contrarily, there are some other treatments in prosthodontics and orthodontics applications which also demand the use of dental wires. In this regard, the wire-appliance for that treatment is still traditionally produced by the handheld operation. Thus, a new and improvised system which could possibly bend the target shape beyond this limitation is required. This refers to a system which is able to perform a more flexible wire bending, not only focusing on the arch-wire.

On the other hand, a robotic device is mostly chosen by most of the reported inventions, which is complex and expensive in spite of the capability of each system to bend only the arch-wire shape. The Suresmile technology, for example, could only bend the basic arch-wire's shape, but not the complex 3D shape in details [9]. Therefore, the potential of using a CNC dental wire bending machine to automatically bend a straight wire into a real 3D target shape equivalent to the virtually designed 3D target shape from the computer is studied. For this reason, it is necessary to consider the adaption of the existing industrial wire bending technology into the dentistry wire bending. Thus, several requirements for the dentistry wire bending machine have been analyzed and the limitations of the existing industrial wire bending technology have been summarized. This information led to the identification of the requirements for an appropriate dental wire bending machine. For example, the machine has to tolerate a small diameter of the wire, a very short feeding length and a sharp bending angle contradicts to the output of the industrial wire bending machine which is more straightforward and mass-produced. In addition, a portable in-office dentistry wire bender is more suitable to be used during the chair-side treatment, contrary to the dimension of the existing industrial wire bending machine which is too large for the dentist's office.

After the design requirements [10] for the wire bending machine have been identified, it is crucial to know how to convert the design information into the CNC machine language. Till recently, a reference to convert the bending points planning data into the machine language is not available. Therefore, a methodology to automatically process and convert the bend points planning data into the CAM data, or the bending code (B-code) has to be established. In order to do so, the most significant wire bending parameters have been initially identified and some mathematical formulae to generate the parameters have been proposed. In this context, the capability of these parameters in controlling the movement of the wire bending machine according to the target shape has been investigated through 3D simulations.

1.2 Objectives of study

- To propose the concept of a flexible dentistry wire bending system.
- To develop a method for the generation of CAM data (B-code).
- To validate the feasibility of the developed method through 3D simulations.

1.3 Significance of research

The present study is necessary in order to understand how the design information could finally be translated into dentistry wire bending parameters (B-code) through some mathematical formulae. The bending parameters (B-code) control the movement of each significant mechanism to realize the wire bending operation until the desired target shape is produced. This provides a general overview how the bend points planning could contribute to the required information for the conversion process. In this study, a theory of 3D linear segmentation is adopted, where the final target is constructed based on multiple 3D linear segments. In this regard, the Cartesian XYZ coordinates of each start point and the end point of every 3D linear segment are employed for the calculation. In addition to this, an automatic vertices coordinate extraction program has been developed in Matlab which could simultaneously extract the XYZ Cartesian coordinates of all 3D points from an Initial Graphics Exchange Specification (IGES) file. This standard format helps to eliminate the barrier of using various kinds of CAD software.

Besides, an automatic B-code generation program has been successfully developed in two different interfaces. At first, Excel is used to develop the first converter as elaborated in [11], but due to the limitation of the interface in indicating the sign of the rotation in a clockwise or counterclockwise direction, the final program has been developed in Matlab, as reported in [12]. The program is capable to process the Cartesian coordinate's data through some mathematical formulae which generate the B-code, in terms of the feed length (L), the rotation angle (β) and the bending angle (θ). Since the calculation does not consider material properties for the compensation of bending angle (θ) and only gives a theoretical value of the bending angle (θ), therefore this study could be further used as a reference for those who might be interested to consider spring back factor in the future.

1.4 Scope of study

The scope of this study is to propose a concept in realizing the automated 3D wire bending machine for dentistry applications. In order to do so, some design requirements for the dentistry wire bending machine have been summarized in [5]. At first, a review of some existing dentistry wire bending systems was carried out and the limitations of each system were observed, as disclosed in [13]-[14]. The review includes an analysis of the

existing industrial CNC wire bending machines in an attempt to adopt the technology into the present study. Details about this review would be discussed in Chapter 2.

As a result of this review, three important mechanisms to realize the 3D wire bending operation have been identified, which are the feeding mechanism, the rotating mechanism and lastly, the bending mechanism. These mechanisms have to be incorporated into the wire bending machine. Then, a theory of 3D linear segmentation algorithm is proposed in [15] which introduces several mathematical formulae to calculate the parameters, in terms of the straight length (L), the rotation angle (β), and the bend angle (θ). The theory considers a multiple 3D linear segmentation of the target shape which resulted in multiple bending to realize the target shape. The XYZ coordinates of the bending point determine the required parameters (L, β, θ) which are used to control the physical movement of the machine. Details about the theory are explained in Chapter 3.

In order to validate the theory, a shape definition analysis was executed by using a sheet metal feature in CAD software, as reported in [16]-[17]. For this purpose, a reverse engineering of the pre-bent target shape was carried out. The target shape was reconstructed in sheet metal feature environment and this allows the simulation of the 3D bending to be executed. Sheet metal feature was used for this purpose since the actual wire bender is not available at this preliminary stage of study and the visual simulation would be impossible. This enabled the visualization of the significant wire bending parameters for the desired target shape. In addition, the generated bending parameters by the sheet metal feature were compared with the calculated bending parameters by using the introduced theory. As a result, the introduced theory was validated as described in [18] due to the satisfactory comparison result between the parameters. Details about this would be elaborated in Chapter 4.

Then, the dentistry wire bending manufacturing workflow is proposed as elaborated in [19], which comprises of data scanning, bend points planning, XYZ Cartesian coordinate extraction, B-code generation and finally the 3D wire bending. A few types of data digitizing have been studied, such as 3D scanning [20], catching the point cloud data through a magnetic sensor [21] and also the capture of 2D teeth image by using a camera [22]. Each type of data digitizing technique produces a different output file, in terms of STL file, 3D point cloud data and 2D image. The variety of these input files have resulted in a different approach of bend points planning, which refers to the planning of the required bend points on the computer. A different procedure is required

to process the data which ultimately aims to extract the XYZ coordinates of all desired bend points. As described above, the XYZ coordinates are crucial for the generation of B-code, which subsequently tells the machine how to perform the wire bending operation accordingly. A brief introduction of these bend points planning methods which are based on the variety of input files in accordance with several case studies would be elaborated in Chapter 5.

Additionally, Chapter 6 describes the developed B-code generation program which automatically converts the design information into the CAM data. This reflects the second objective of this study, which is to develop a method for the generation of CAM data (B-code). In addition to this, an automatic XYZ coordinate's extraction program from the IGES file is also introduced, as stated in [12]. Details about these programs would be explained in Chapter 6.

In order to achieve the last objective of this study, which is to validate the feasibility of the developed method, 3D simulations of the generated B-code were executed, as elaborated in [23]. Chapter 7 examines this matter by using 3D point cloud data as the input data, in addition to the 2D images. Some experiments to collect the 3D point cloud data with the use of a magnetic sensor would be described through a few selected test models. Then, a few programs to process the 2D image file to prepare the required data in accordance with the manufacturing workflow would also be discussed. Result for the 3D simulation of each case has demonstrated the capability of the introduced B-code program to establish the reliable code for the 3D wire bending operation. This simultaneously validates the proposed mathematical formulae for the feed length (L), the rotation angle (β) and the bend angle (θ).

Subsequently, Chapter 8 introduces the mathematical definition of human dental arch forms for both labial and lingual arch curve, in an attempt to design an optimized dental arch shape for the orthodontics treatment. This could possibly improve the theory of 3D linear segmentation bending into a more semi-optimized curve, especially for the arch-wire treatment planning. However, the actual arch-wire bending operation to realize the semi-optimized arch shape has not been studied, which remains as the recommendation for the future work which would be discussed in Chapter 9.

CHAPTER 2

AUTOMATED DENTISTRY WIRE BENDING SYSTEMS

2.0 Preface

The purpose of this chapter is to present a comprehensive review of published literature investigating various methods and techniques for the automated fabrication of dental wire in orthodontics and prosthodontics applications. Formerly, the wire bending operation for the wire-related treatment in dentistry application was manually done in laboratory production. The traditional method is used around the world until a few innovations in the computer-aided manufacturing of the wire bending emerged to automate the orthodontics arch-wire bending. The application of these reported advancements is only focusing on the orthodontic treatment, while the other wire-related treatments in prosthodontics and orthodontics application are still traditionally performed in the hand-made operation. Therefore, the present chapter reviews the bending procedure of each invention to automatically bend the wire, in an attempt to adopt these emerging technologies for the concept of a flexible dental wire bending system. In addition, the existing industrial wire bending machine technology is also explored, the future development trend is foreseen, and the limitations of each system are briefly discussed.

The review surveys the key publications in English related to dentistry wire bending from the past three decades, as reported in [14]. Several patents and research articles are briefly examined to identify the direction of this research area in the future. In general, several attempts to automate the dentistry wire bending operation have been reported in [31]-[52]. However, all of these inventions are related to orthodontics arch-wire manufacturing for braces treatment in orthodontics application. Generally, there are two approaches in the braces treatment, one is based on a flat arch-wire and customized brackets, and the latter approach is based on a customized arch-wire and off-the shelf brackets [13], as depicted in Figure 1. The use of dental CAD/CAM to create individualized bracket system is another emerging research topic [24], but this chapter only focuses on the automated dentistry wire bending systems to bend both a flat arch-wire and also the customized arch-wire for these two approaches.

(a) (b)

Figure 1: (a) flat arch-wire (b) customized arch-wire

CAD/CAM technology refers to digital design and manufacture, where CAD software recognizes the geometry of an object while CAM software is used for the manufacture [25]. The development of CAD/CAM in dentistry is based on three elements [26], namely: (a) data acquisition, (b) data processing, and (c) manufacturing. In this study, the development of dental CAD/CAM in the field of dental wire manufacturing has been reviewed by collecting relevant findings from academic journals and patents. In reality, CAD/CAM for dental restorations is more explored in comparison to dental wire bending. However, the future research in dental wire bending is evolving. Taking a lead from the restorative dentistry, the future fabrication of lingual orthodontic wire is predicted to be performed by an in-office robot [27]. This has become a reality after the Suresmile technology was introduced for lingual and labial wire treatment planning and manufacturing in orthodontics. Other than that, the potential applications of these technologies might greatly expand. Therefore, the following sub-sections discuss those automated systems, differentiating between the use of a robotic device and also the CNC dental wire bending machine.

2.1 Orthodontics arch-wire bending robot

The use of robot to replace human in the dentistry wire bending could reduce the labor dependency and medical expenses tremendously. The advances of robot in orthodontics have been discussed in general by [6],[28], and [29] including the application of robot in manufacturing of complete or partial denture, dental implantology, and the bending of arch-wire. Since the focus of the review is broad, thus there are a few inventions in the field of arch-wire bending which are omitted. Therefore, a more detailed

review with regard to the use of a robot in the arch-wire bending has been conducted, adding to some unreported inventions to the above-cited works.

Orthodontics has taken a lead in the technological advances, driven by the increasing numbers of patients who require the reduction of treatment time, minimal post-operative pain and enhanced periodontium [30]. In the first invention by [31], a robot consists of two characteristic design features; a bending cone that could move forward and backwards to bend the wire, and a rotating cone that could twist the wire is reported. This robot was developed by the predecessor of the assignee of Suresmile technology [32]. The robot was manufactured and sold as part of complete orthodontic solution by Geyer Medizintechnik GmbH in Berlin, Germany, but never widely commercialized. According to [33], the Bending Art System (BAS) is the first computer-aided device for the automatic bending of the orthodontic individual arch-wire, consists of an intraoral camera, a bending device and a computer program. This system composes of three main elements namely the advancing apparatus, the torsion apparatus, and the bending apparatus. However, the patent would not work for many orthodontic prescriptions and was rather imprecise because the robot has no effective feedback mechanism for detecting how the wire in fact was bent after a particular bending or twisting operation was performed. The deficiencies in this patent have led to the development of a more improvised system, known as the Suresmile technology [34].

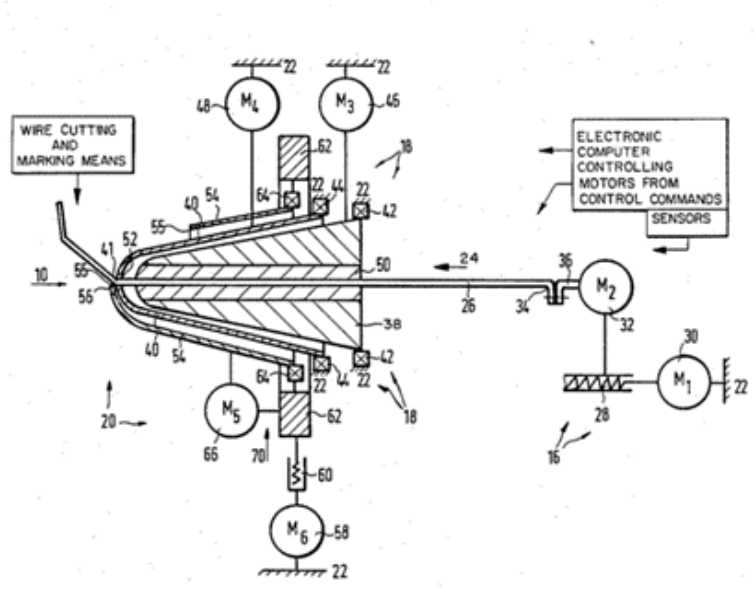


Figure 2 : First wire bending system by Orthuber [31]

The Suresmile technology was patented in 2003, which consists of two grippers [35]. One of the grippers is being mounted to a six-axis-robot arm and thus moveable, while the other gripper is designed to hold the arch-wire during the bending operation [36]. The gripping tools are incorporated with force sensors [37], which are used to determine the required over bend [38] to get the desired final shape of the arch-wire. The manufacturing process uses straight pieces of wire and step by step applies bends and/or twists to the wire, thus performing an arch-wire [39]. The Suresmile robot is capable of bending fully three-dimensional orthodontic arch-wires. A few modifications to the initial patent were reported in [36]-[40], which have several shortcomings. Firstly, the geometry of the bends that can be manufactured is limited and does not allow for creation of all the wire shapes desired by orthodontists. In cases where the bend length must be very short, such as in lingual cases, the treatment options can be limited by this approach. Secondly, highly accurate arch-wires are difficult to obtain in a repeatable manner by the robot described in these patents. The final patent, as disclosed by [32] has overcome these drawbacks, thereby providing a further substantial advance of the prior art. The adoption of this technology towards the braces treatment is reported in the United States, Germany, Canada and Australia [41].

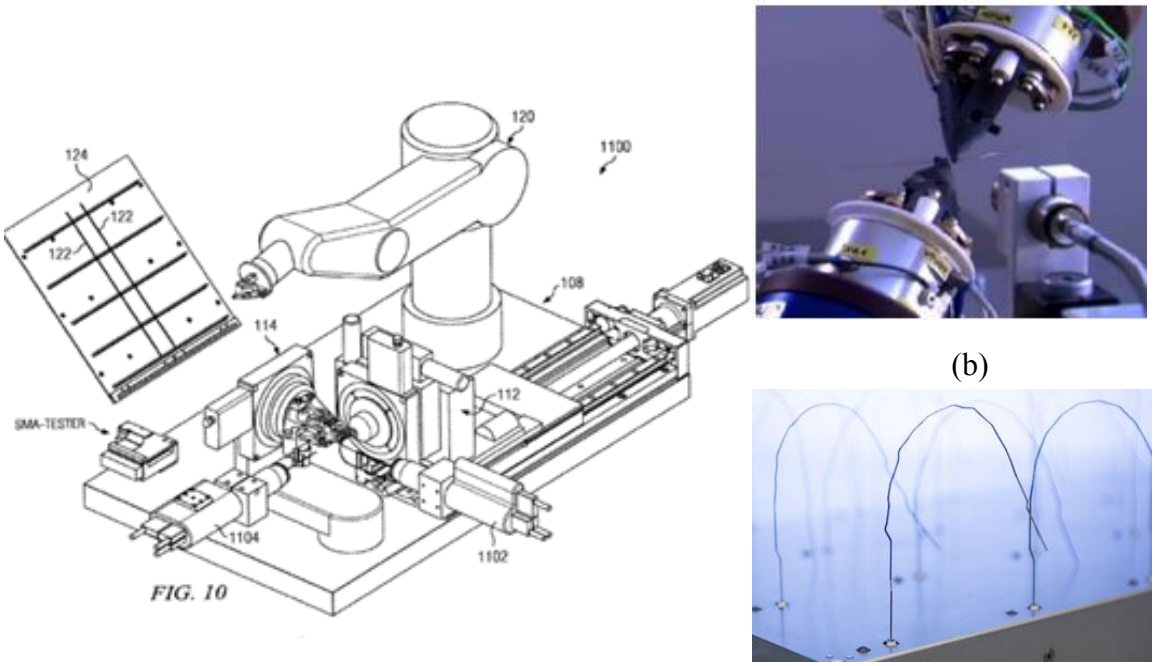


Figure 3 : (a) Suresmile robot (b) the grippers (c) the customized arch-wire

The applications of Suresmile technology include the labial treatment and also the lingual treatment, as shown in Figure 4. On the other hand, the procedure of this technology consists of three steps. Firstly, an initial 3D model scanning of the teeth is conducted using digital scanning technology. Secondly, the treatment planning stage is carried out with the use of 3D virtual simulations. Thirdly, a customized arch-wire is produced by the robot. The Suresmile robot uses a movable manipulator that can move relative to the fixed gripping tool about 6 axes and is able to form arch-wires with any required second and third order bends quickly with high precision. The robot is however, can bend only the basic arch-wire's shape, but not the complex 3D shape in detail. Nevertheless, the Suresmile robot had successfully proven in reducing treatment time than conventional approaches and improved the quality [42].

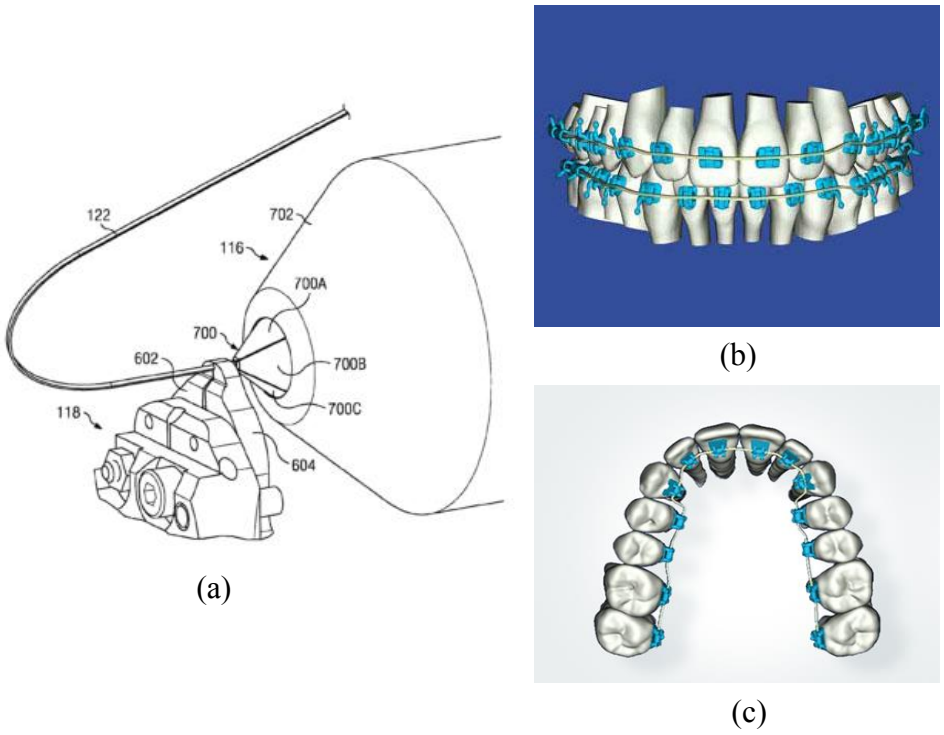


Figure 4 : (a) Suresmile wire bending operation (b) labial treatment (c) lingual treatment

There is another commercialized robotic arm by Sirona Dental GmbH in Bensheim, Germany which is capable to plan the bracket system [43] and also bend the arch-wire [44]. The bracket planning treatment by this Orthorobot is patented, but the arch-wire bending system is not reported. Sirona Dental GmbH claims in the website that the individual arch-wire is bent by the robot. This is another commercialized robot which is developed by the German and the system is already being used in practice. There are

three cornerstones to the Orthorobot lab process. Firstly, the virtual planning of bracket positions is executed to guarantee the best possible bracket position for each tooth. Then, the brackets are bonded onto the model in precisely the desired position in accordance with the computed target coordinates. However, the subsequent transfer of the brackets to the patient is carried out by the indirect bonding method, with the use of transfer tray. The indirect bonding method is one of the limitation of this system. Thirdly, using the saved target coordinates, the production of customized arch-wire is done by the robot. In addition, Orthorobot is designed to bend a precise and customized individual arch-wires for lingual treatment. However, this system is only capable to bend the first order bends.

(a) (b) (c)

Figure 5: (a) planning of bracket positions, (b) bracket bonding, (c) customized arch-wire

Another method and an apparatus for bending orthodontic wires were introduced by [45]-[46]. The system was designed for lingual orthodontics and the author claimed the invention has substantial improvements over the other cited applications [47],[35],[31]. Inventors for this patent are also one of the developers of the Suresmile technology.

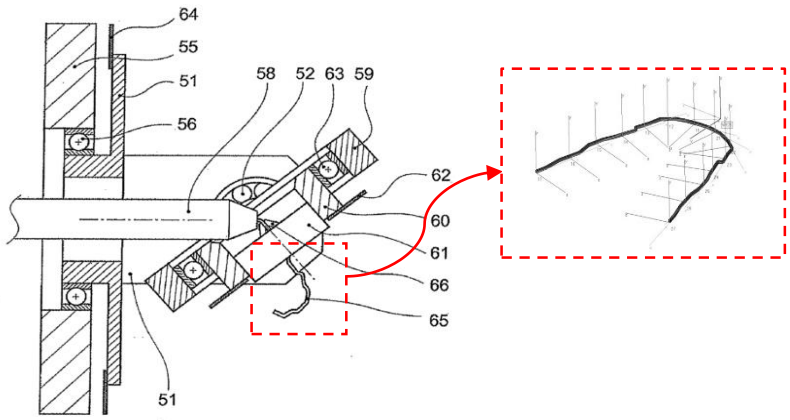


Figure 6: Schematic diagram of the apparatus by Rubbert and Weise [45]-[46]

Another invention for lingual orthodontics was developed by Gilbert and known as the Lingual Arch-wire Manufacturing and Design Aid (LAMDA). LAMDA is a polyarticulated robotic system for the production of lingual arch-wires through the use of an anterior plate and controlled movement of a plurality of posterior bending units. With this system, the orthodontist is presented with a view of the dental arch on a computer screen, and design a virtual lingual arch-wire as a series of straight lines to the screen image of the dental arch. The system comes with a software, also named as LAMDA which calculates the length of each of the straight line and also the angle between each pair of adjacent section. The input for this system is a 2D image of the patient's teeth where the software uses pixels as a unit of measurement. The gantry robot, however can only realize the movement in XY plane which refers to 2D wire bending. As a consequence, it cannot bend the arch-wire with closed loop and the application is very limited [48]. The system is however suitable to be used during chair-side treatment due to the small size of the in-office robot.

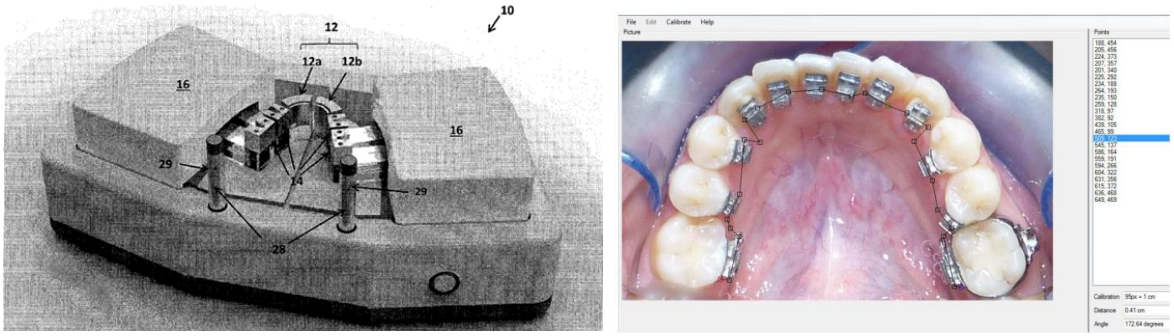


Figure 7: (a) LAMDA gantry robot, (b) treatment planning

Other than previously cited inventions, there are another two reported advancements related to arch-wire bending robot by researchers from Harbin University Science and Technology, China. However, these two systems are still at theoretical research and preliminary experiment level. They have proposed a prototype of the Cartesian type arch-wire bending robot for a specific labial arch-wire bending [49], in addition to robotic arm known as MOTOMAN UP6, as reported in [50]. The prototype for both systems is indicated in Figure 8.

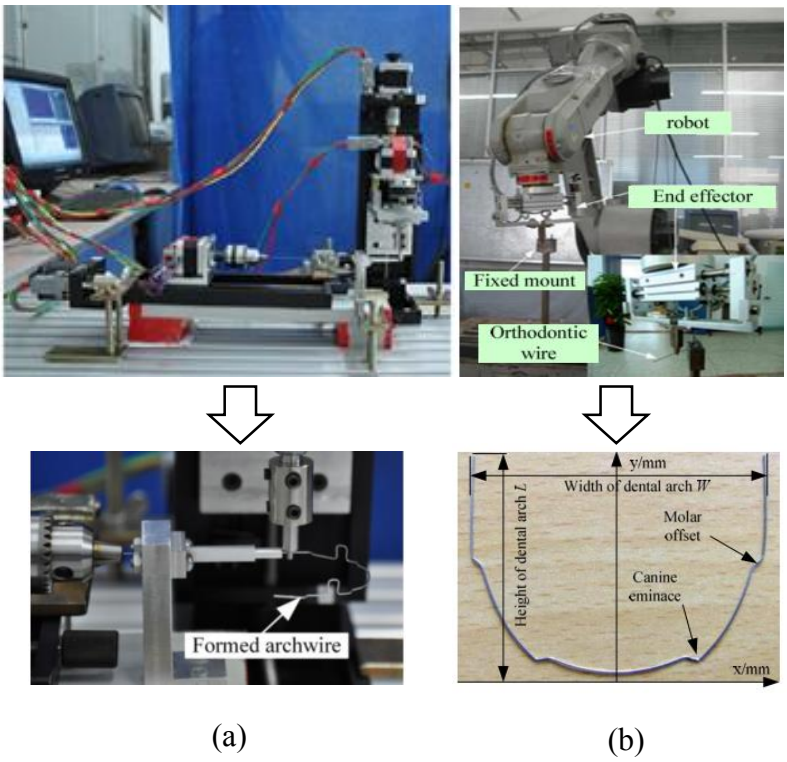
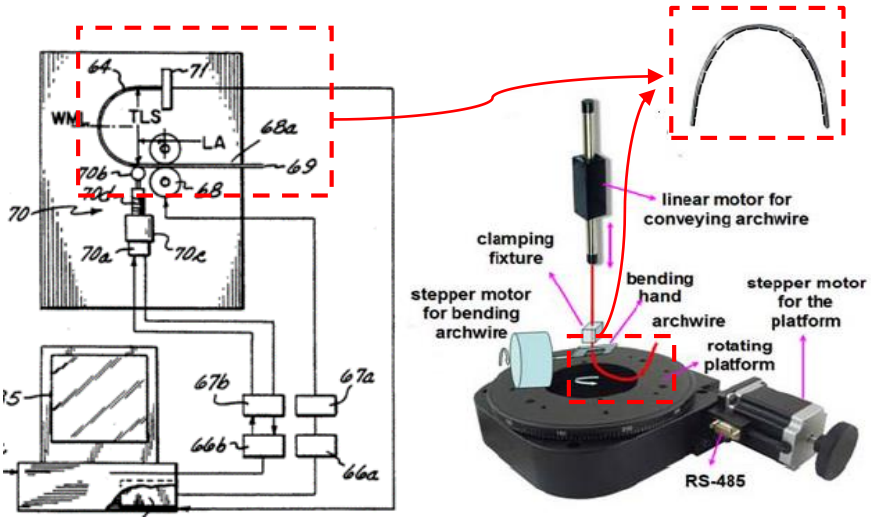


Figure 8: (a) Cartesian robot [51], (b) MOTOMAN UP6 [50]

2.2 CNC dentistry wire bending machine

Other than the arch-wire bending robot, there are also some introductions to CNC dental wire bending machine to bend the arch-wire. The first patent describes an apparatus that takes a straight arch-wire and imparts a simple planar arcuate curvature to the wire [47],[52]. The wire is customized in the sense that the shape of the arc is designed for a particular patient, but the wire bending apparatus is limited to a customized bracket system which requires a flat arch-wire. Therefore, the introduced system could only bend a flat arch-wire, not a customized arch-wire. In particular, the wire bending apparatus cannot produce any complex bends in the wire which require a combination of translation and rotational motion [32],[45]. This patent has been frequently cited by the other reported inventions. On the other hand, the second invention was introduced in 1993, as discussed in [9]. It is known as Automated Bending Arch-wire Machine (ABAM) and similar to the previous patent, it is only capable to bend a flat arch-wire. Also, there is no subsequent updates concerning to this invention which makes it remains at theoretical research and preliminary experiment level.



(a) (b)

Figure 9: CNC wire bending machine by (a) Andreiko [47],[52], and (b) Zhang [9]

2.3 Industrial CNC wire bending machine

In this study, the possibility of adopting the industrial CNC wire bending technology into the dentistry wire bending was also examined. A brief review of the existing industrial wire bending machine has indicated a few limitations. First, the wire bending in dentistry is rather complicated and requires high degree of accuracy to ensure perfect fixation to customer's mouth which could not be performed by the existing technology. The industrial wire bending, on the other hand, is designed for a simpler shape, contrary to the target shape in dental wire bending which is more specific and complicated. Secondly, the review has shown that the existing industrial wire bending machines are designed to bend thick wires, contrary to the needs in dentistry applications which demand a thinner dental wire in the range of 0.18 mm -0.29 mm [6]. Thirdly, the size of the existing industrial wire bending machine is not appropriate to be used during the chair-side treatment. Therefore, a portable in-office CNC dental wire bender is more relevant. Lastly, all of the commercialized machines from different developers come with their own built-in interface, which automatically changes the design information into the wire bending language. Each machine has its own method for this conversion process and there is no general information disclosed to the public concerning to this conversion process. Consequently, the present study introduces a theory which could be used for this computer-aided manufacturing (CAM) data generation without considering material

properties in the calculation. For this reason, the conversion formulae and the feasibility of the theory would be demonstrated through a few examples. This could be a reference for other researcher in the future and simultaneously promote the development of prosthodontics, orthodontics, and relative fundamental theory, in general.

Thus, a few strategies which might be possible to bend the dentistry wire has been brainstormed, in reference to the existing industrial wire bending technology. The first strategy (feed and bend) reconciles a 2D wire bending technology, with a wire feeding mechanism is attached together to a wire bending mechanism. The wire is being fed automatically and go through a couple of concentrically rollers which act as a holder and a bender. It is important for a bending device to hold the wire in stationary position during the bending process while the bending head performs the wire bending operation. The bending head rotates clockwise and counterclockwise during the operation, with regard to the desired target shape. In this context, rollers are used to bend the wire and the degree of bending depends on the size of the rollers. Bending is influenced by several factors, like the size of bending head, the number of bending pins (more is better for a faster process) and the layout of the bending head. A small bending angle requires a small size of roller and vice versa. The bending head material also plays a significant role in terms of the lifetime. Hard tool steel could be used to ensure a long wear life. Unfortunately, the strategy only allows a 2D output.

The second strategy is similar to the (feed and bend) strategy, with an addition of rotation (feed, bend and rotate). In this strategy, a 3D wire bending is allowed either by including a wire rotating device which secure one ends of wire and allow rotational movement of wire, or by adding the bending head degree of freedom. In the first strategy, the bender rotates in clockwise and counterclockwise only, but in the second strategy it can move in a *z* direction. Therefore, a 3D bending is possible. The difficult part in this strategy is the arrangement of the wire bending mechanism to avoid unexpected clashing between the bent wires and bending head. A small radius could be bent by theoretically attaching a small roller head but the stability of such bend towards the wire stiffness has to be taken into consideration.

Despite a few limitations offered by the existing industrial wire bending machine to realize the dentistry wire bending operation, the review has encountered basic strategies which should be incorporated into the concept of a CNC dental wire bender. Firstly, a sending mechanism, or could also be referred as the wire feeding mechanism must be

established. For this mechanism, there are another few options, like the wire feeding mechanism could either automatically feed the wire or the wire has to be manually measured and cut before it is fed into the feeding mechanism. For the dentistry applications, the second option is more appropriate due to the accuracy. Secondly, the wire could either be rotated for the changing of working plane, or an additional rotating mechanism has to be equipped into the machine. Without this rotating mechanism, only a 2D wire bending could be realized. Finally, the bending mechanism is the final significant mechanism which has to be considered to realize a flexible wire bending operation. Table 1 summarizes the findings.

Table 1: 2D and 3D wire bending machine strategies

Strategy	*Concept*
Feed and Bend	2D wire bender
Feed, Bend, Rotate	3D wire bender

2.4 Summary of dentistry automated wire bending systems

Germany is leading in developing an advanced robot for an arch-wire bending compared to other developed countries. Orthodontic planning treatment which employs a robot for bracket planning and individual arch-wire bending has been offered in Germany through Suresmile technology and Orthorobot. Recently, China has started to develop their own arch-wire bending robot and also the CNC bending machine. For the use of robot, there are two basic design requirements to be considered, which are the planning of the arch-wire spring back and the planning of bending point to realize the accurate arch-wire bending [6]. Research difficulty in the arch-wire bending robot system are the spring back analysis, the introduction of bending algorithm, and the manufacturing of personalized orthodontics arch-wire. In the bending process, the moving distances of orthodontic arch-wire bending robot are relatively short, but the requirements for movement accuracy is high [49]. However, with the development of correlation technique and theory, such as new structure, and sensor and control theory, the robot will be widely applied in prosthodontics and orthodontics in the future, despite the complexity of wire bending principles and methods. In order to relate this with the development of CNC wire bending machine, the highlighted criteria was studied and would be further discussed.

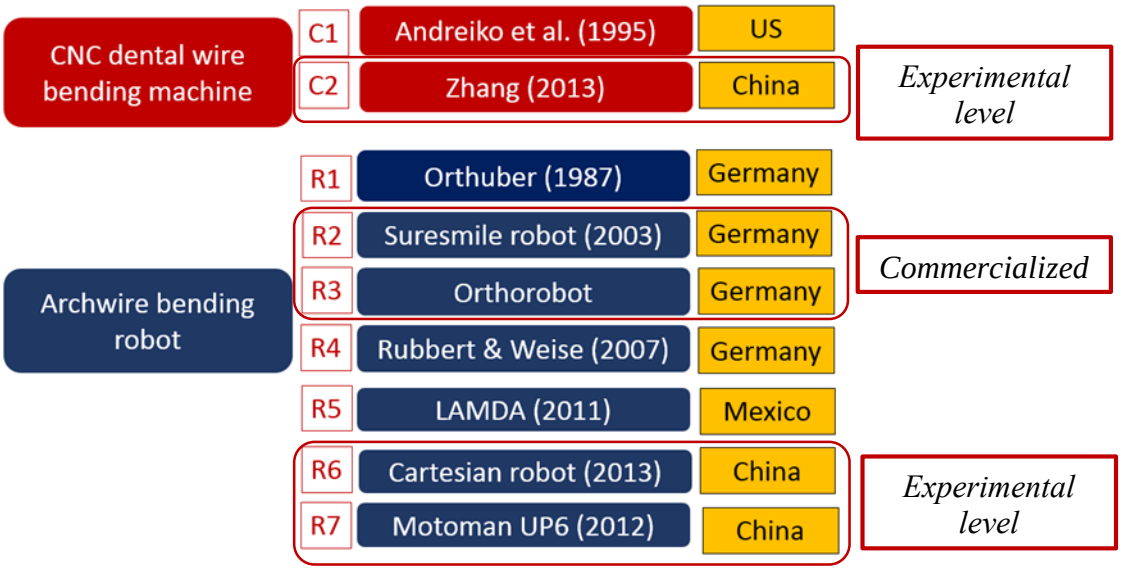


Figure 10: Summary of the reported inventions

Technology advancements in dental CAD/CAM have successfully offered a better manufacturing of dental restoration several decades ago. The same evolution for dentistry wire bending has started, with the recent focus on arch-wire manufacturing. Figure 10 shows the summary of the reported inventions. Simultaneously, the former technology has constantly improved. For instance, the development of world's smallest 3D dental scanner by the Austrian Institute of Technology has indicated a high level of miniaturization is no longer impossible which motivates the present study in proposing the concept of an in-office wire bender. The review highlights the significant mechanisms which have to be embedded into the system in order to realize a 3D wire bending. Even though arch-wire bending robots have generated more interests among researchers and some introduced systems are already available for commercialization, CNC wire bending machine is preferred due to the low-cost and less complex than employing a robot. In addition, the intended application is focusing on the in-office use during the chair-side treatment which makes a desktop size and a straightforward machine operation are somehow preferred. Surveys in digital dentistry have indicated the acceptance of recent technology among dentists are intermediate where the old and established dentists are less prone to changes due to the difficulty of changing their conventional method and also the reluctance in learning a new knowledge. This implies how important a straightforward and easy wire bending system is, in order to make it more acceptable by dental experts in the future.

CHAPTER 3

THEORY OF 3D LINEAR SEGMENTATION ALGORITHM

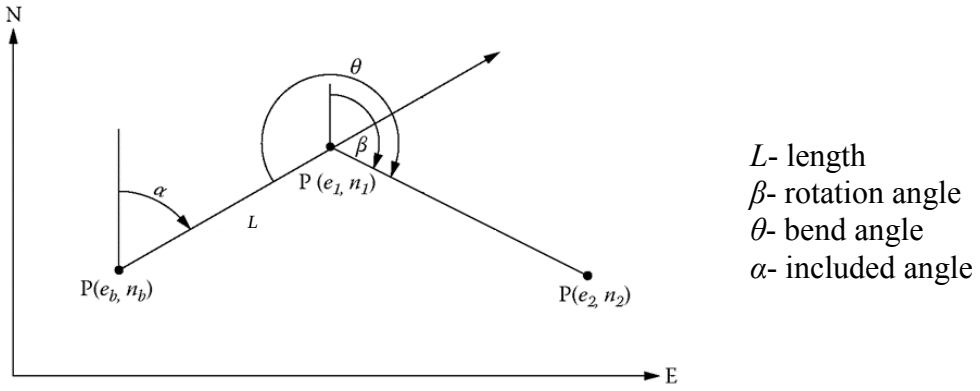


Figure 11: Relative spatial parameters

Digital wire design in orthodontics and prosthodontics is complicated due to freeform shapes which make them difficult to be predefined generically [54]. As a result, the conversion process from CAD to CAM data would be impossible with these freeform shapes. Therefore, a 3D linear segmentation algorithm is proposed in the present study, where the freeform line would be presented by multiple 3D linear segments, before mathematical algorithms are employed to generate the B-code. In this theory of 3D linear segmentation, a freeform shape is firstly segmented into multiple straight line segments. The whole target shape is composed of a sequence of multiple 3D linear segments in order to be bent easily by the CNC wire bender. The bending of 3D real shape is made based on the relative spatial parameters (L-the length, θ- the bend angle, β-the rotation angle) of the adjacent 3D linear segments. The following sections would discuss the definition of each parameter, in addition to the related mathematical formulae to calculate them.

3.1 Algorithm for the feed length (L)

The straight length (L), can be calculated through XYZ coordinate values of its two endpoints. For the entire equation, X, Y, and Z refer to the Cartesian coordinates of the two endpoints in the 3D linear segments. Therefore, for a 3D representation, the

vector-valued function is used to represent those lengths in the vector form. According to Figure 12, Point 1 is an example of a 2D point representation, while Point 2 represents a 3D point. In this context, a vector representation for **S1**, referring to Point 2 and the origin as the two endpoints is as follows:

$$\overrightarrow{\boldsymbol{S1}} = (x_2\text{-}x_1)\,\mathbf{i} + (y_2\text{-}y_1)\,\mathbf{j} + (z_2\text{-}z_1)\,\mathbf{k} = m_1\mathbf{i} + n_1\mathbf{j} + p_1\mathbf{k} \quad (1)$$

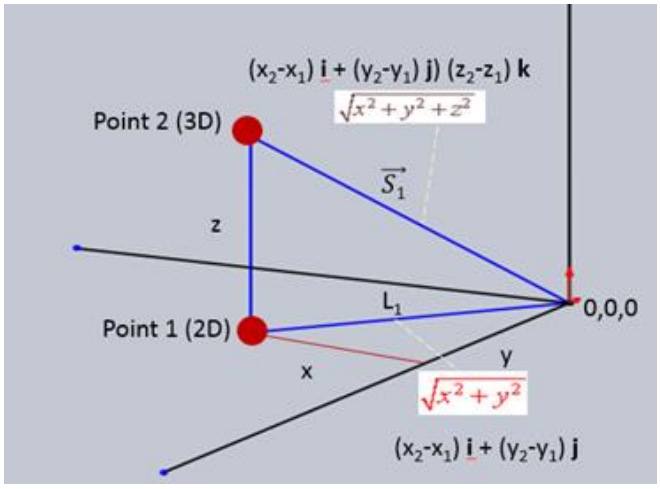


Figure 12: Feed length (*L*)

3.2 Algorithm for the rotation angle (*β*)

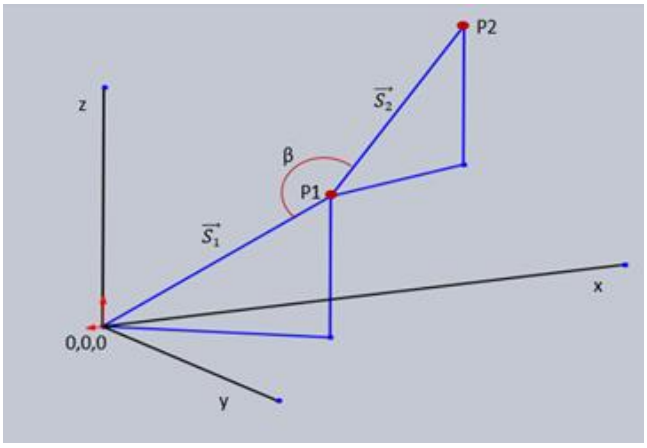


Figure 13: Graphical illustration of the rotation angle (*β*)

Since every line segment represents a 3D spatial line segment, therefore it is usually not in the same plane. In this context, there is a rotation angle (*β*) on the space between the adjacent 3D linear segments. The included angle between the adjacent planes

is referred as the rotation angle (β) between its adjacent segments. Thus, the algorithm for the rotation angle (β) is represented by Equation 2 which has been adapted and expanded from [49].

$$\cos\beta = \frac{\overrightarrow{S1}\overrightarrow{S2}}{|\overrightarrow{S1}|.|\overrightarrow{S2}|} \tag{2}$$

$$\cos\beta = \left| \frac{A_1A_2 + B_1B_2 + C_1C_2}{\sqrt{A_1{}^2 + B_1{}^2 + C_1{}^2} \cdot \sqrt{A_2{}^2 + B_2{}^2 + C_2{}^2}} \right|$$

where;

$A_1 = (y_i-y_{i-2})\,(z_{i-1}-z_{i-2})-(y_{i-1}-y_{i-2})\,(z_i-z_{i-2})$

$B_1 = -\,(x_i-x_{i-2})\,(z_{i-1}-z_{i-2})-(x_{i-1}-x_{i-2})\,(z_i-z_{i-2})$

$C_1 = (x_i-x_{i-2})\,(y_{i-1}-y_{i-2})-(x_{i-1}-x_{i-2})\,(y_i-y_{i-2})$

$A_2 = (y_{i+1}-y_{i-1})\,(z_i-z_{i-1})-(y_i-y_{i-1})\,(z_{i+1}-z_{i-1})$

$B_2 = -\,(x_{i+1}-x_{i-1})\,(z_i-z_{i-1})-(x_i-x_{i-1})\,(z_{i+1}-z_{i-1})$

$C_2 = (x_{i+1}-x_{i-1})\,(y_i-y_{i-1})-(x_i-x_{i-1})\,(y_{i+1}-y_{i-1})$

3.3 Algorithm for the bend angle (θ)

$$\cos\alpha = \frac{\overrightarrow{S1}\overrightarrow{S2}}{|\overrightarrow{S1}|.|\overrightarrow{S2}|} \tag{3}$$

In this calculation, the included angle (α) has to be initially acquired before the bend angle (θ) could be consecutively obtained. Hence, Equation 3 is used for calculating the included angle, α. In this approach, the included angle is calculated first according to the information obtained from the 3D linear segments. Then, the required bending angle, θ could be obtained by adopting Equation 6. In this equation, **S1** is calculated in accordance to Equation 1, while **S2** could be derived using the same approach, as follows:

$$\overrightarrow{\mathbf{S2}} = (x_3-x_2)\,\mathbf{i} + (y_3-y_2)\,\mathbf{j} + (z_3-z_2)\,\mathbf{k} = m_2\,\mathbf{i} + n_2\,\mathbf{j} + p_2\,\mathbf{k} \tag{4}$$

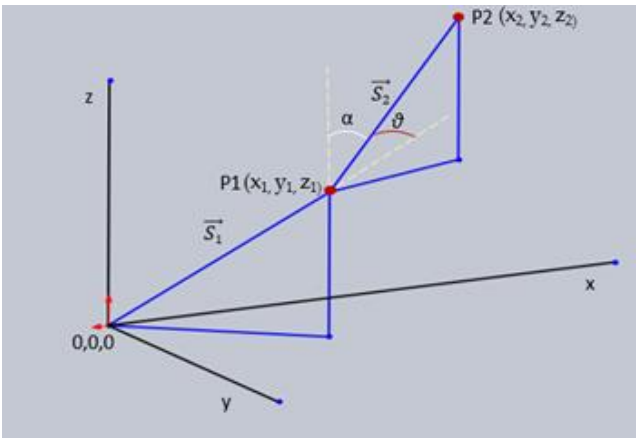


Figure 14: Bend angle (θ) and the included angle (α) between adjacent linear segments

Therefore Equation 3 is generalized into the following equation;

$$\cos\alpha = \left[\frac{m_1 m_2 + n_1 n_2 + p_1 p_2}{\sqrt{{m_1}^2 + {n_1}^2 + {p_1}^2} \cdot \sqrt{{m_2}^2 + {n_2}^2 + {p_2}^2}} \right] \quad (5)$$

where;

$m_1 = (x_{i+1}-x_i)$; $n_1 = (y_{i+1}-y_i)$; $p_1 = (z_{i+1}-z_i)$

$m_2 = (x_{i+2}-x_{i+1})$; $n_2 = (y_{i+2}-y_{i+1})$; $p_2 = (z_{i+2}-z_{i+1})$

Finally, the bend angle, (θ) could be calculated through Equation 6, as follows:

$$\theta = 180° - \alpha \quad (6)$$

CHAPTER 4

THEORY OF B-CODE SHAPE DEFINITION ALGORITHM

This is the second stage of the study after the review of automated wire bending system was completed, where a dentistry target shape is used to understand the wire bending operation, in relation to the bending parameters. Sheet metal feature is adopted since the CNC mechanism is still developing and the real simulation process to validate the introduced theory is impossible. The objectives of this analysis are to establish the significant wire bending parameters (B-code) through 3D linear segmentation approach, and at the same time to validate the result of this calculation by using sheet metal feature. The purpose of this study is to show that the proposed procedure could work well with sheet metal feature and if it is, the B-code could be applied to the CNC wire bender. Figure 15 shows the general overview of this work.

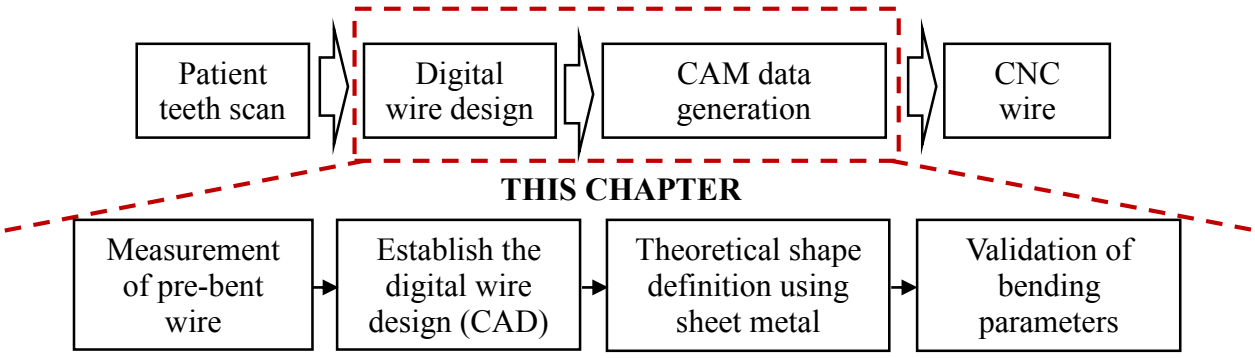


Figure 15: General CNC dental wire bending process and the highlight in this chapter

4.1 Definition of B-code parameters

The tool path algorithm serves as a data front-end for a CNC milling. Similarly, an automated wire bending system, unfortunately cannot actually bend a wire without a special algorithm, required to create the CNC instructions (commonly known as G-code) for bending. The B-code forms a critical components for wire bending. Therefore, the algorithm that is capable of processing the input file resulting in a sequence of the wire bending generation is developed which finally produces the B-code for the system.

The CNC code generation program produces four parameters in accordance with the designed mechanisms, in terms of the wire straight length (L_{before}), the wire feeding length (L_{after}), the plane rotation angle (β) and also the bending angle (θ). A few mathematical formulae have been adjusted and adopted to the proposed system to compute those L, β and θ values. In this program, the unit of (L) is in millimeter (mm), while the angle of (β) and (θ) are described in degrees. However, the program could also produce the output in radian, depending on the preference. However, the unit of degrees would be employed in this report. In addition, the output of (β) indicates a positive (+) and a negative sign (-), which refers to clockwise rotation and counterclockwise, respectively.

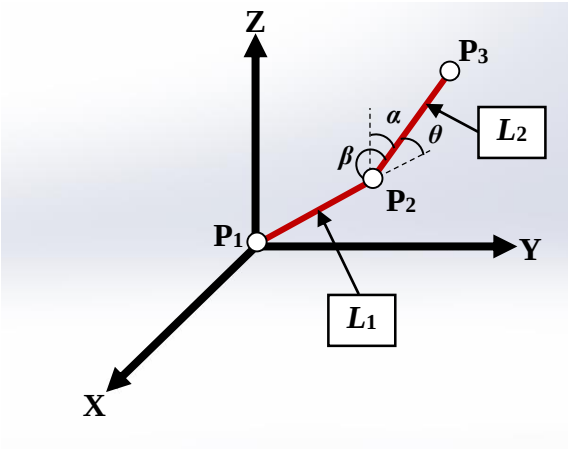


Figure 16: Definition of bending parameters between adjacent 3D linear segments; P: 3D points (P_1, P_2, P_3), L: length (L_1, L_2): θ: bend angle, α: included angle, β: plane rotation angle

This section explains some mathematical equations, employed in the study for the conversion process. According to the introduced 3D linear segmentation theory, Equation 1 is further expanded into Equation 7 in order to compute the unbend length (L_{before}). In a wire bending operation, a straight wire is fed to the bending mechanism. The value of this straight length is defined as L_{before} in this work. In this formula, X, Y and Z refer to the coordinate of start point and end point for each line segment, where root sum square is adopted in calculating the length, or the magnitude of the position vector.

$$L_{before} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \quad (7)$$

Subsequently, the inserted straight wire length (L_{before}) would be bent at the desired bend point according to the calculation of the feed length, (L_{after}). This feed length (L_{after}) is a result of the straight length (L_{before}) subtracted to the tangent of the bend angle, as written in Equation 8. In short, the feed length depends on the bend angle (θ) and also the bend radius, r. The zero value of this bend radius, r produces an equivalent length for L_{after} and L_{before} between the line segments, indicating the previously described straight length in the 3D linear segmentation algorithm.

$$L_{after} = L_{before} - \left(r \times \tan\left(\frac{\theta}{2}\right)\right) \tag{8}$$

In the meantime, Equation 2 from the previous section is utilized in calculating the rotation angle (β) through vector analysis. In this equation, X, Y and Z are all referring to the vertex coordinate of the line segment, either pointing to the vertex before, or after the present vertex. The formula considers two planes which refers to the current working plane and the previous working plane. In this case, the first bend does not have the previous working plane which resulted in a zero value of β. In this calculation, firstly, the vector ($\mathbf{P}$, $\mathbf{Q}$) perpendicular to the working plane is calculated using the cross vector. Secondly, the length of this vector which is perpendicular to the working plane is calculated using the root sum square ($\|\mathbf{P}\|\|\mathbf{Q}\|$). This step is repeated for both planes. Finally, the rotation angle between these vectors, β is estimated. On the other hand, the direction of rotation from the previous plane to the current plane is also considered. In this consideration, counterclockwise direction is given as negative, while the clockwise direction is given as positive. Lastly, Equation 3 to Equation 6 describe the 3D vector approach used to compute the bending angle (θ). In this calculation, firstly, the vector of each adjacent line segment ($\mathbf{v_1}$, $\mathbf{v_2}$) is calculated and subsequently, the length of these vectors are computed through the root sum square ($\|\mathbf{v_1}\|\|\mathbf{v_2}\|$). The included angle, α which refers to the angle between these vectors is finally obtained through the dot product of ($\mathbf{v_1}$, $\mathbf{v_2}$).

4.2 B-code shape definition through sheet metal feature

In order to proceed, a reverse engineering of the labial bow was executed as the target shape. Initially, the measurement of the labial bow was executed and the

measurement was used for the establishment of the 3D wire model in the sheet metal feature. Subsequently, the simulated relative spatial parameters from the feature in term of (L, β, θ) were compared with the calculated B-code parameters for validation. In reality, there is an introduction to CAD/CAM technology in the fabrication of the base plate through stereolithography (SLA) machine [55], but the labial bow is still prefabricated manually. The purpose of the analysis was to understand the target shape by using the physical dental appliance which was bent manually by the technician, with a consideration of several type of features in dentistry application, such as curve, loop and also the first order bend.

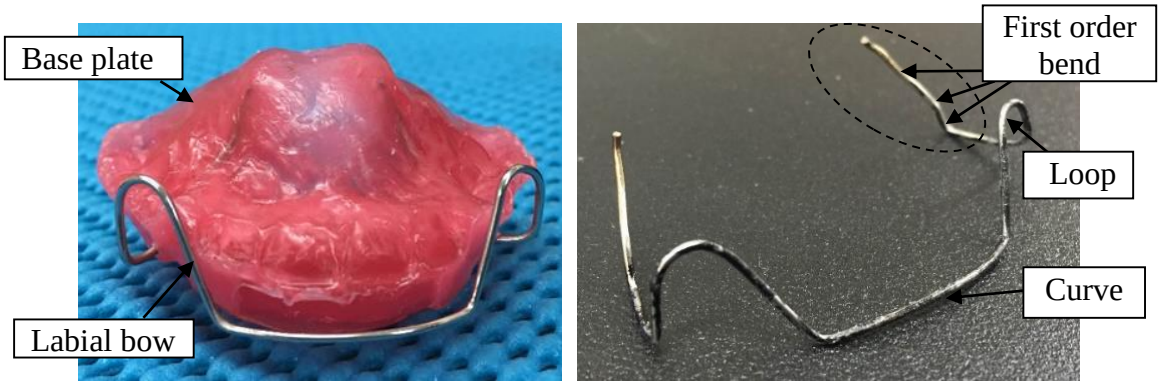


Figure 17: Physical model of the target shape wire (labial bow); (a) in the real application, while (b) indicates some type of bend in dentistry considered in the analysis

In this example, the first order bend section was sketched approximately to the original shape. On the other hand, the loop and the curve were constructed from multiple 3D linear segments, in accordance with the theory of 3D linear segmentation. These features were re-constructed in CAD by segmenting the freeform shapes with regard to the theory of 3D linear segmentation. To perform a linear segmentation of a parametric curve P (u), $0 \leq u \leq 1$, the real parameter u is increased by an amount, which is not constant but to make sure it follows the curve closely so that a set of points $P_0 = P(u_0)$, $P_1 = P(u_1)$, $P_2 = P(u_2)$ …, $P_n = (u_n=1)$ approximating P(u) is generated, as referred to [56].

$$u_0 = 0$$

$$ui+1 = ui + \Delta u$$

The 3D linear segmentation algorithm approximates the parametric curve into a set of lines and the algorithm converts them into smaller linear motions within real time,

known as B-code. The segmentation, or the chord error is not discussed in the present work, rather than to highlight the theory of converting CAD to CAM data with this approach. In reality, dental wire parts are produced by a sequence of bending operations. The bending process starts with a feed of straight wire and ends up with a 3D object of interconnected flanges. The bending operations are planned to be executed on a CNC wire bending machine. The visual simulation of this bending operation is significant as it helps to define the target shape, estimate the bending sequences and ultimately understand the know-how about the bending process. The simulation also helps to determine the important parameters during the bending operation. This is significant in generating a process plan for the CNC machine, which refers to the identification of suitable sequences to perform the different bend operation. The bending sequences contain some geometrical bending features and are determined on the basis of a target shape design. It should be sequenced so as to avoid part–tool, part– machine and part–part collisions during the real bending operation [57].

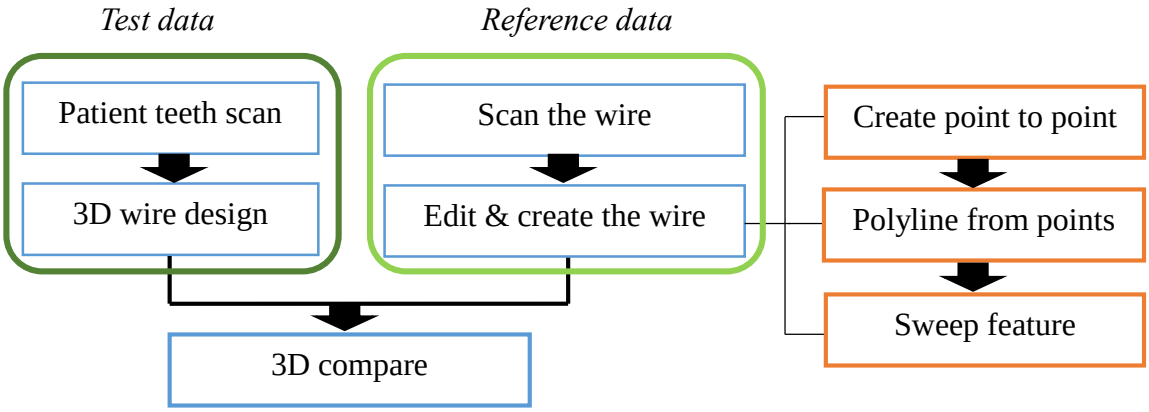


Figure 18: Procedure to create the reference and test model

Figure 18 describes the process of test data creation in CAD (sheet metal feature) and reference data creation by means of Reverse Engineering (RE). Details for both data generation would be briefly explained in the following section. In addition, a 3D comparison for both data was executed in Geomagic environment which generated a color mapping result. The purpose of this comparison is to verify the wire design, whether it could be created as close as the actual wire measurement or not. A color mapping approach was chosen since it is the most straightforward and simple for a verification method related to geometry [58].

4.2.1 Reference data creation

3D scanning is normally used to digitize the required shape which is subsequently employed to construct a digital 3D model. In this study, a Makerbot digitizer was utilized for the scanning of an orthodontic retainer while a MakerWare for Digitizer created the industry standard STL file. MakerBot Digitizer Desktop 3D Scanner creates a digital 3D model of a physical object by taking a rapid sequence of pictures as the object rotates on the MakerBot Digitizer turntable. Firstly, laser lines were created with two lasers that outlined the profile of the object, while a camera concurrently took a series of photographs. Those lasers were mounted on the left and right corners of the MakerBot Digitizer and the scanning process employed one laser after another until the whole image was generated. Secondly, MakerWare for Digitizer software took the photographs of the laser lines and combined them to create the point cloud data of the digitized object. Then, a MakerBot MultiScan™ Technology was executed for the scanning improvement where the MakerWare for Digitizer rescanned the wire from a different angle and combined the two sets of scan data.

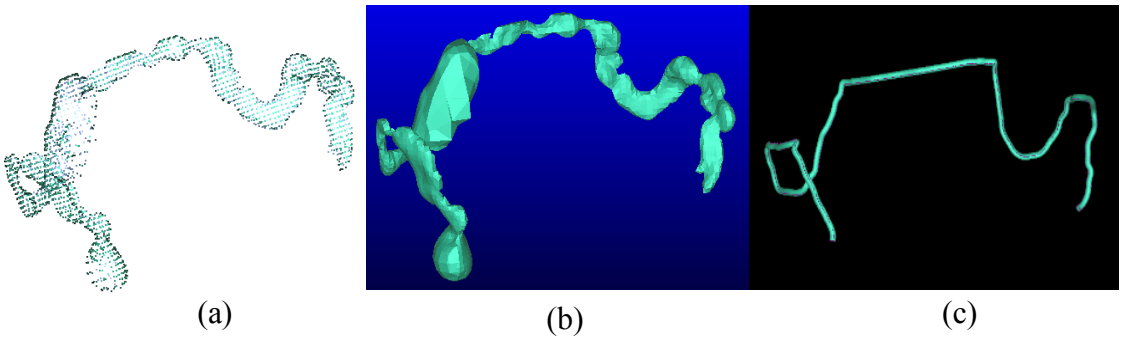


Figure 19: (a) point cloud data (b) mesh data (c) created wire

Afterwards, the editing of the STL file was performed in Geomagic where the deletion of unwanted noise near the desired shape was executed and a mesh of the point cloud data was generated. The raw scanned data acquired the feature of the target shape, but the appearance was not so good for the subsequent 3D comparison. Therefore, in order to increase the accuracy of the 3D comparison result, a new shape was created based on the scanned image. In order to do so, several points were selected according to the STL file. Later, a curve through points was generated with a b-spline and a sweeping tool was applied to make it resembled a wire, with a diameter of 0.5 mm. The diameter was chosen

in regard to the actual reference wire.

Alternatively, a second reference data was created by using a probe digitizer, by digitizing the wire in the *z* direction. Previously, a laser type digitizer was used to generate the scanned model but due to the wire reflectivity [59], the generated model was insufficient. Hence, another digitizer was utilized which created the following 3D model. Figure 20 exhibits the generation of a scanned wire which was edited in the ProEngineer for the comparison purpose. Initially, several points were created on the raw scanned data. After that, a line through point option was used to connect all the points chosen earlier. In the end, a sweep feature was used to make the line looks like a wire, with the same actual wire diameter.

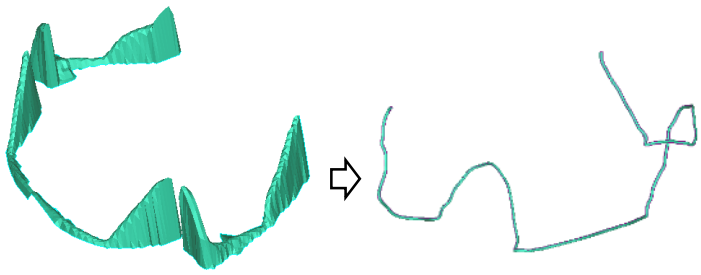


Figure 20: Transformation of the scanned wire model to a reference model

4.2.2 Test data creation

In the real application, a straight wire is bent with undetermined parameters that depend on several factors, like the feature of the bending plier, the method of bending and most importantly, the expertise of bender. Normally, the bender could not envisage the final outcome before the operation or foresee the consequence of diversifying the bending parameters. However, with a sheet metal feature, CAD model is converted directly to sheet metal model and the sequence of bending could be visualized in a 3D design environment. The modifications of some bending parameters are also possible to foresee the effect of changing some parameters to the designed shape. Therefore, to what extent the 3D sheet metal model could represent the real target wire was investigated in the present study.

Sheet metal parts constitute a very specific type of solid model. The wall thickness is constant throughout the model and the bends are applied using a selected bend radius value. The fundamental sheet metal feature is a flange. Solidworks sheet metal has four

different types of flanges that can be used to create parts. The flanges add material of predefined thickness in different ways. In this study, the base flange was employed to create the base feature for the sheet metal part. The added bending parameters are editable and could be modified according to the preference of users. Therefore, the bending procedures required for the target shape would be discussed, elaborating on the bending sequences, bending points and also the bending parameters applied for each section.

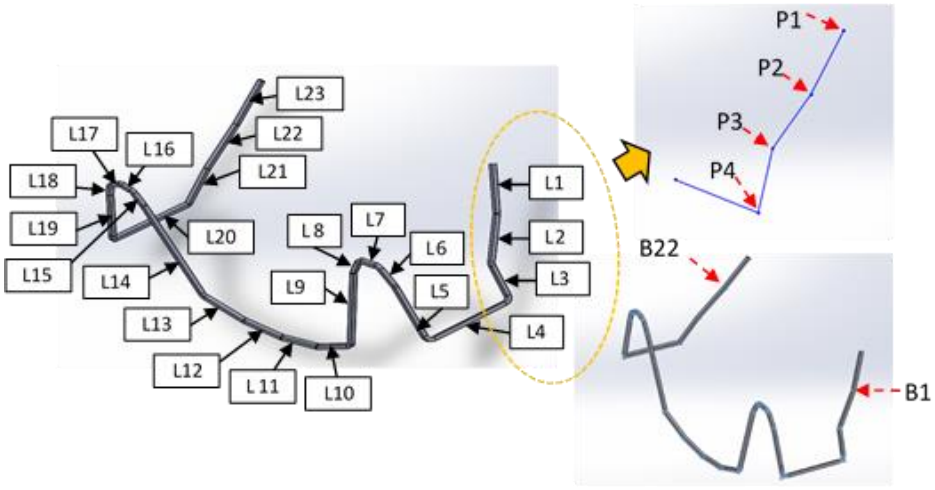


Figure 21: Sheet metal model of 23 line segments (L1-L23)

According to Figure 21, the target shape consists of 23 linear segments with variety of length sizes. Each line segment has its start and endpoint which is referred as *P* in this example. The number of line segments is approximated according to the pre-bent physical wire. In this approximation, the freeform shapes are segmented into multiple 3D linear segments. The discretization is performed in order to apply the mathematical formulae based on the 3D linear segmentation theory in calculating the desired bending parameters (L, β, θ). This is clearly demonstrated in the figure where in the front section, multiple line segments are used to create the curve in the middle (L10-L13), the loop for right (L6-L8) and left (L15-L18) sides. These line segments are sketched on some inclined planes, according to the real model. From this sheet metal simulation, the number of bending required to produce this target shape has been digitally acquired after the feature is applied to the sketch, which are 22. Based on this simulation, we found out that the radius of bend, *r* plays a significant role in shaping the bent length (L_{after}), as theoretically described by Equation 8. Furthermore, *r* value is proportional to the unbent length (L_{before}). A short unbent length (L_{before}) has to be accompanied by a smaller bend radius *r*, otherwise the bending operation is impossible due to negative bent length (L_{after}). To relate this finding

with the bending mechanism, the size of the bend pin which is designed to perform the bending operation would determine the minimum bent length (L_{after}) which it could work on. In a practical point of view, this is important criteria as the dental wire often deals with a very short bent length in between bend. Therefore, the unbent length (L_{before}) or it could also be defined as the feeding length has been considered as one of the important parameter for this bending operation. The input of unbent length data (L_{before}) into the CNC machine tells the feeding mechanism how long a straight wire to be fed into the bending mechanism before any bending operation starts.

4.3 Validation of B-code parameters

Table 2: XYZ coordinates for each 3D point along 23 linear segments

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**	**P12**
X	8.06	9.88	11.5	13.23	12.95	10.26	8.66	6.05	4.55	0.27	-5.39	-10.86
Y	-2.96	-5.2	-6.49	-9.88	-2.81	2.69	4.3	4.24	2.57	-6.67	-8.3	-8.77
Z	-33.32	-28.07	-22.44	-19.19	-10.77	-15.48	-16.88	-16.92	-15.57	-7.95	-6.77	-6.56
	P13	**P14**	**P15**	**P16**	**P17**	**P18**	**P19**	**P20**	**P21**	**P22**	**P23**	**P24**
X	-16.33	-22.06	-27.36	-28.42	-29.8	-31.4	-31.95	-33.91	-33.75	-32.58	-31.26	-29.9
Y	-8.41	-7.15	1.33	2.55	2.87	2.44	1.24	-4.39	-11.07	-9.62	-8.66	-7.22
Z	-7.06	-8.31	-15.61	-16.67	-16.98	-16.67	-15.68	-11.03	-18.46	-23.1	-28.99	-34.65

In this section, an algorithmic approach to evaluate the bend was executed based on the theory of 3D linear segmentation. In this procedure, the objective is to verify the introduced theory by a comparison of parameters data (L_{before}, L_{after}, β, θ) between the simulated sheet metal data and the calculated data. Even though a sheet metal feature might have a different algorithm in its bending calculation, the target is to show the results. The values of unbent length (L_{before}), bent length (L_{after}), bend angle (θ), and plane rotation angle (β) were extracted manually from the sheet metal feature design. Then, the coordinate of each point was extracted and compiled in Table 2. These coordinates were employed for calculating the parameters data (L_{before}, L_{after}, β, θ) by using the introduced theory. In this calculation, the value of bend radius, r was referred to the sheet metal setting. The comparison results, as shown in Table 3 have demonstrated a close proximity of reading. Therefore, the introduced formulae are considered reliable and could be further used in the B-code generation program.

Table 3: Comparison of bending parameters ($L_{before}, L_{after}, \beta, \theta$)

Length, L (mm)							Bend Angle θ (degrees)		Rotation Angle β (degrees)	
Line	L_{before} (SM)	L_{before} (theory)	Bend location	(r)	L_{after} (SM)	L_{after} (theory)	(θ) (SM)	(θ) (theory)	(β) (SM)	(β) (theory)
L1	6.00	6.00	1	1	5.91	5.90	10.00	9.98	0	0
L2	6.00	6.00	2	0.5	5.77	5.77	31.99	31.95	179.88	179.88
L3	5.00	5.00	3	1	3.91	3.91	86.99	86.97	179.99	179.99
L4	11.00	11.00	4	1	9.05	9.05	90.00	90.01	0.04	0.04
L5	7.72	7.72	5	0.5	6.65	6.65	16.61	16.49	90.34	90.34
L6	2.67	2.67	6	0.5	2.34	2.34	53.39	53.57	0.47	0.47
L7	2.62	2.62	7	0.5	2.11	2.10	54.75	54.61	0.14	0.14
L8	2.62	2.62	8	0.5	2.29	2.29	15.25	15.28	0.15	0.15
L9	12.72	12.72	9	1	12.18	12.18	50.86	50.81	179.82	179.82
L10	6.00	6.00	10	1	5.4	5.41	14.30	14.35	0.06	0.06
L11	5.50	5.50	11	1	5.27	5.27	11.43	11.40	0.46	0.46
L12	5.50	5.50	12	1	5.3	5.3	10.89	10.92	1.11	1.11
LI13	6.00	6.00	13	1	5.46	5.46	47.53	47.50	0.62	0.62
L14	12.38	12.38	14	0.5	11.91	11.91	8.02	7.92	179.97	179.97
L15	1.93	1.93	15	0.5	1.72	1.72	38.80	38.88	0.2	0.20
L16	1.46	1.46	16	0.5	1.11	1.11	36.35	36.12	0.28	0.28
L17	1.68	1.68	17	0.5	1.27	1.28	51.83	52.24	0.14	0.14
L18	1.65	1.65	18	0.5	1.39	1.39	4.61	4.45	1.09	1.09
L19	7.56	7.56	19	0.5	7.06	7.06	88.00	87.92	88.88	88.88
L20	10.00	10.00	20	0.5	9.23	9.22	60.00	60.01	0.06	0.06
L21	5.00	5.00	21	0.5	4.71	4.68	8.00	7.96	179.84	179.84
L22	6.11	6.11	22	0.5	6.09	6.06	5.00	4.93	179.41	179.41
L23	6.00	6.00			5.98	5.98				

4.4 SCAN-TO-CAD comparison

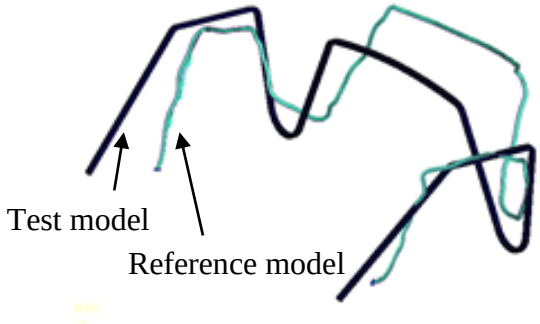


Figure 22: Best fit alignment between the reference and test model

This section describes the difference between the scanned model and the drawn sheet metal model in Geomagic environment. The purpose of this analysis is to randomly check the deviation between the constructed 3D model in sheet metal feature and the actual physical wire, which was digitized through 3D scanning devices. In order to do so,

a 3D compare tool was used for the comparison analysis where it generates a 3D, color-coded mapping of the differences between the reference object and the test object. The reference object is the object that is selected in the model manager. The test object is chosen from a dropdown list in the 3D compare dialog. In addition, the test object can be points, polygon, or CAD while the reference object can be polygons or CAD. Therefore, in this study, the reference object of the scanned image was wrapped as a polygon data, while the sheet metal model was chosen as the test object. A polygon data was chosen over the CAD data for the reference object since the CAD and polygon models act as very different reference objects. When an IGES-CAD model is toleranced with the points, the software goes back to the CAD engine to get the exact deviation of the point. However, once the model is converted to a polygon that information is lost and the points are toleranced to a faceted polygon model. Figure 22 shows the alignment between these models before a 3D comparison was subsequently conducted.

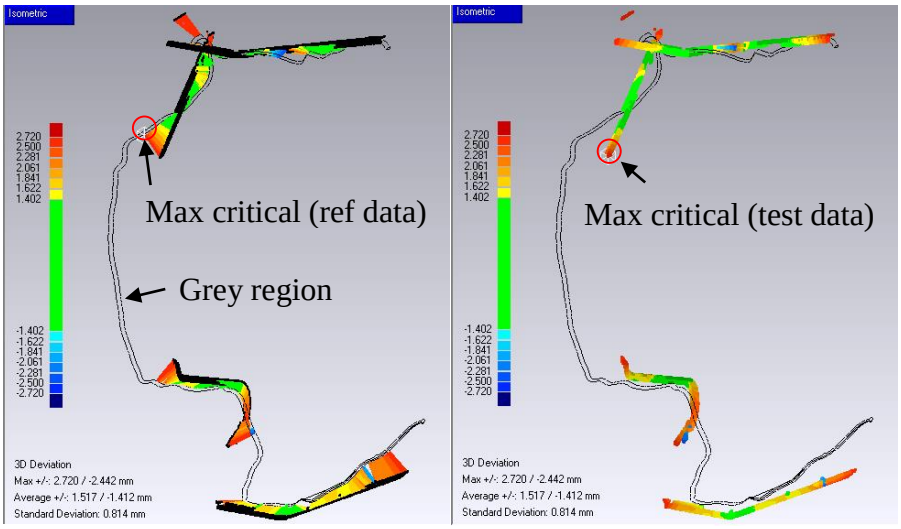


Figure 23: Color mapping result

Figure 23 exhibits the color mapping result of this comparison on the reference object. This operation generates a color-coded spectrum model illustrating areas of correspondence and deviation between the two surfaces. The divergence is obvious, with some portions displayed in gray. The gray region shows insufficient data for valid comparison to the reference object. Areas of the resulting spectrum model that appear green illustrate regions of highest correspondence and in this study, deviate less than ± 1.402 mm from one another. By default, surfaces at the higher end of the spectrum

(yellows and reds) highlight areas in which the second model has positive relief, i.e., it is larger than the original. Those surfaces colored at the lower end of the spectrum (blues and violets) highlight areas of negative relief, or places where the second model is smaller than the original. The red circle shows the most critical point for the alignment.

4.5 Summary of B-code shape definition analysis

As a result of this study, the introduced mathematical formulae based on the 3D linear segmentation theory have been validated through the adaption of sheet metal feature. The formulae incorporate the Cartesian XYZ coordinates which have been expanded from the 3D vector approach. The theory considers the XYZ Cartesian coordinates for the start point and endpoint of the 3D line segment resulted in some important bending parameters in terms of (L_{before}, L_{after}, β, θ). The parameters are intended to control the movement of the CNC wire bender, similar to the G-code. The unbent length data (L_{before}) informs the feeding mechanism on the straight length of wire to be fed into the bending mechanism, while L_{after} indicates the feed length of each bending operation. This information is delivered to the bending mechanism where it tells the start point for each bending operation. On the other hand, the bending angle (θ) controls the bending mechanism in generating the right degree of bend in between two adjacent 3D line segments. In addition, the plane rotation angle (β) commands the turning mechanism, attached to the bending mechanism on the relative rotation of plane between two adjacent bending. This is necessary if the working plane of the subsequent bend changes, in the case of 3D non-planar bending. This is a theoretical work to demonstrate how the bending parameters for the CNC wire bender could be established, with no consideration of the material properties.

CHAPTER 5

PROPOSED DENTISTRY WIRE BENDING MANUFACTURING WORKFLOW

In this chapter, the dentistry wire bending manufacturing workflow which is proposed in this study would be elaborated. Figure 24 exhibits the traditional dental wire bending manufacturing workflow. In this conventional approach, the highlighted box indicates a repetitive bending for each bend point, until the desired target outline is successfully obtained. Normally, a dental technician will mark the desired bend points on the wire in reference to the pre-designed wire path on the dental cast model. Then, the bending operation will be executed subsequently using human hand and some specific pliers. This manual wire bending operation heavily relies on the expertise of the wire bender and in this wire bending operation, the bender has to make multiple bends at each bending point due to wire spring back effect. The pre-bent wire has a tendency to return to its initial position due to this factor and this demands the bender to make multiple bends to compensate the effect. Also, the bender has to frequently check the produced bending angle in accordance with the designed feature line for a maximum accuracy.

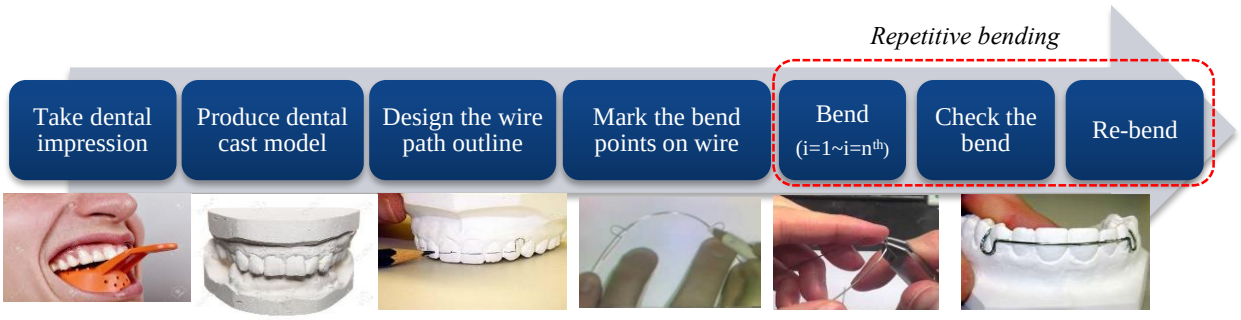


Figure 24: Traditional wire bending manufacturing workflow

Additionally, this can simultaneously increase the bending time due to some major adjustments and leads to bender fatigue. Furthermore, it has a huge tendency to create errors, thus affecting the efficiency of the wire for the specified treatment. Cooperation between experienced dental specialist and the skilled technician is often needed to produce high-quality dentures with a low revision rate. Therefore, the automated wire bending

system is aiming to resolve all of these concerns and improve the existing practice among dental technicians. Figure 25 illustrates the general overview of the proposed wire bending manufacturing workflow in this study, in consideration to several case studies.

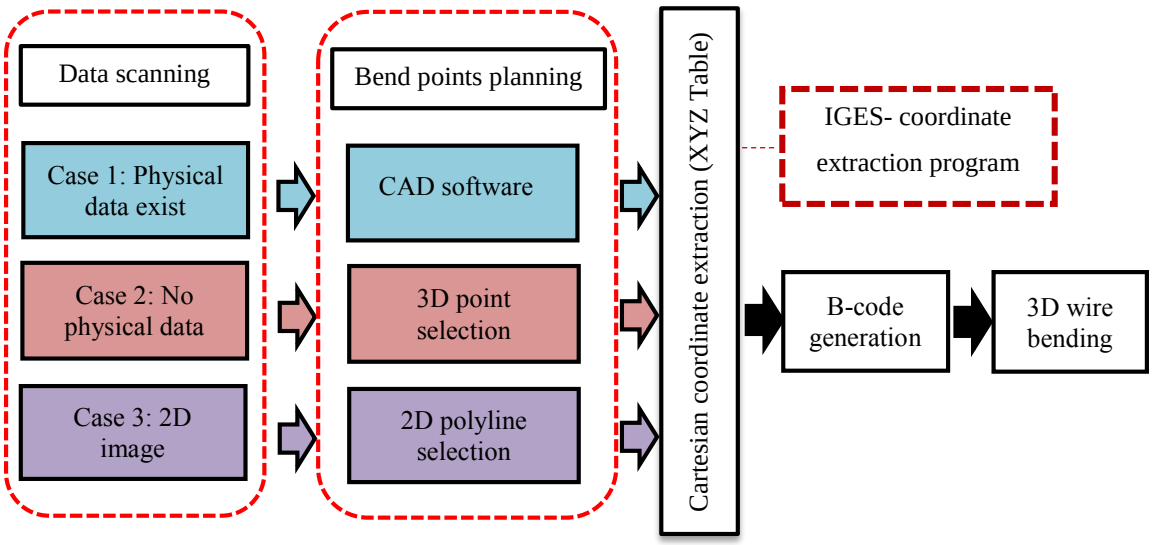


Figure 25: Proposed wire bending manufacturing workflow

In case 1, the physical data is available in terms of the dental wire appliance or the dental cast, as shown in Figure 26. In this context, the 3D scanning of this physical data could be executed through 3D scanner to establish the digital data. Then, the bend points planning towards this digital data could be done on the scanned dental cast or by referring to the scanned wire appliance in CAD, depending on the final objective. For example, if the reverse engineering of the wire appliance is planned to be executed, then the bend points planning could be done in reference to this scanned wire appliance. On the other hand, if the physical dental cast of the patient is available, the following bend points planning could be directly performed on the STL file of the dental cast. Details about the bend points planning and the XYZ extraction from the IGES file would be subsequently explained.

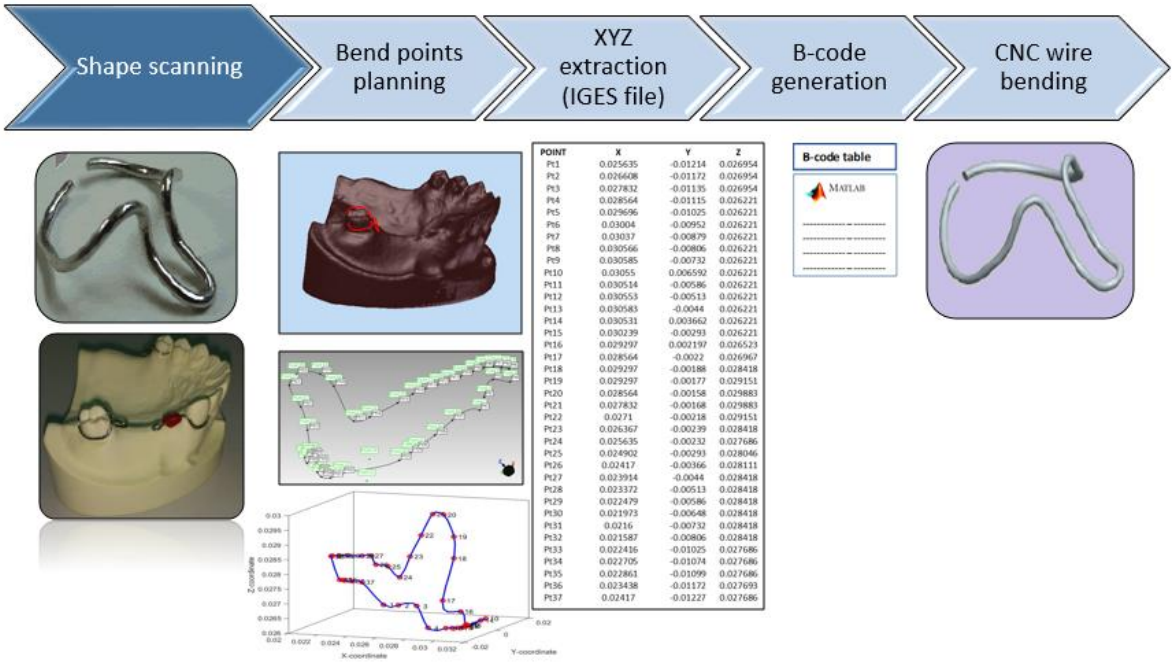


Figure 26: Case 1- Existence of physical data

Secondly, in case if the physical data is not available, a magnetic sensor is adopted to acquire the desired feature line, which resulted into a collection of 3D point cloud data. This type of data scanning requires some programs to process the data, as briefly summarized in Figure 27. The reason of experimenting with this method is to allow the acquisition of the desired target shape from the dental impression to be traced and recorded digitally. In a manual wire bending operation, a dental technician draws the desired wire outline with a pencil on top of the patient's dental cast. Subsequently, the manual wire bending operation begins in reference to this approximated line and each time the wire is bent, the accuracy of the bending angle will be checked in reference to this outline for every single bend until the whole target shape is successfully produced. In order to automate the wire bending operation, the target shape could be digitized in reference to the scanned teeth or the approximated line on the dental cast. This type of data acquisition produces a dataset of 3D points along the digitized contour, including noise. In consequence, the identification of the preferred points for the B-code generation has become more challenging. These noise frequently belong to objects which surround the test sample being digitized, such as fixtures, measurement table, or some other part of the assembly to which the digitized part belongs. However, in the case of non-contact methods, such laser triangulation, those points can originate from objects located further

away [60]. To some extent, the noise can also be the result of measurement errors (due to operator errors, system-specific errors and/or errors due to specific nature of the digitized object, some external disturbance as vibrations) et cetera. Those noise have to be eliminated in order to maintain the quality of surface reconstruction, which lead to the introduction of data filtering program. Moreover, the bend points planning and the XYZ coordinate extraction for this type of input data would be consecutively executed in the 3D point selection program.

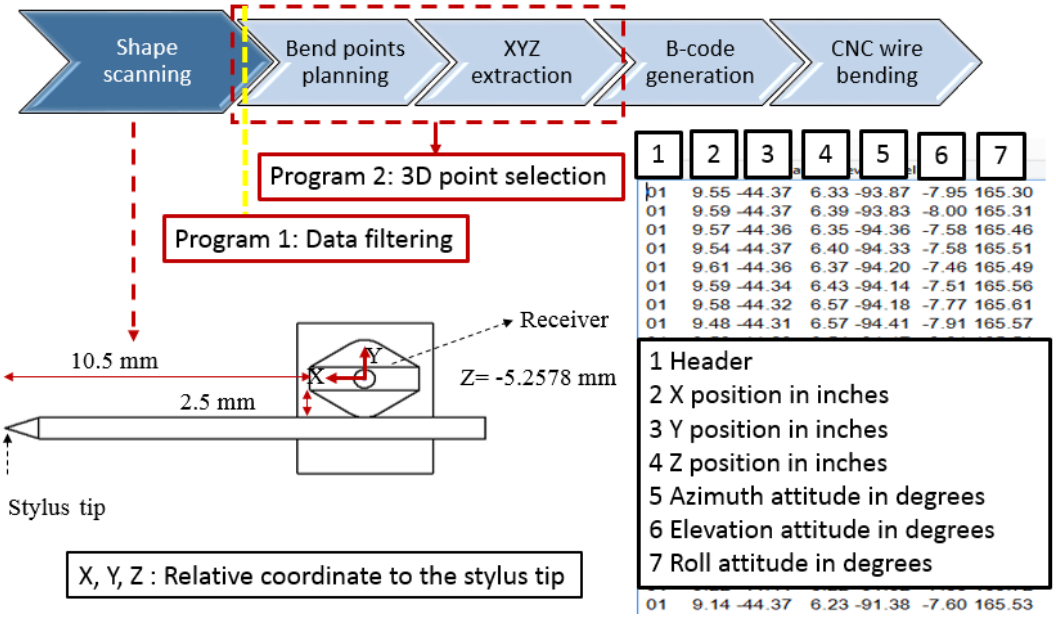


Figure 27: Case 2- Non-existence of physical data

Finally, in case of a 2D image file is considered, some other programs which could process this kind of information have been explored, as indicated in Figure 28. Similarly, the bend points planning and the XY coordinate extraction process could be simultaneously done in the 2D point selection program. The (X, Y) location coordinates of each pixel representing landmarks of interest would be highlighted and converted into a matrix of numerical data. These numerical data represent the location coordinates of each selected point. This is how the XY coordinate of the bend point could be translated from the 2D image through the introduced program. This case is normally applied for the treatment planning of the lingual treatment, for example, which allows the dental expert to estimate the appropriate lingual arch-wire for a patient through a 2D teeth image.

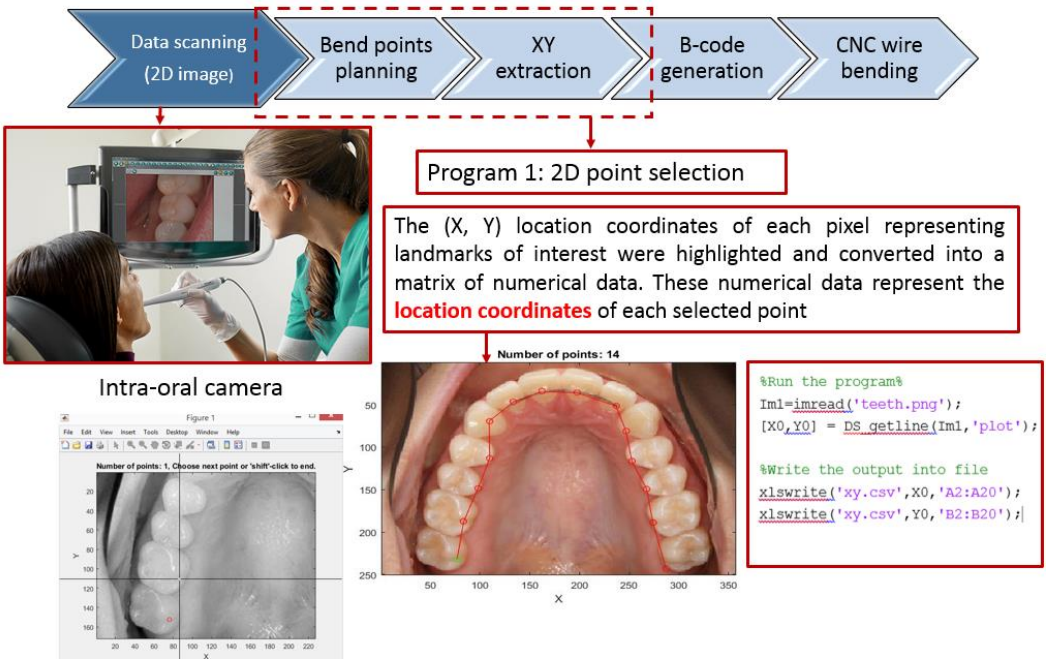


Figure 28: Case 3- Consideration of 2D image file

5.1 Data scanning

In a regular dentistry treatment, the patient teeth scan would be performed by a dentist and afterwards use to help analyzing the needs of the patient [61]. An advancement in the field of intra-oral scanning system has made it possible for the creation of the patient's 3D teeth model during the chair-side treatment, using an intra-oral scanner device with an opened-system, such as Trios 3 (3Shape). Manipulation of the scanned data with a software program (CAD) to create a virtual model is an example of digital geometrical data exchange in the CAD/CAM technology [62]. The opened-system, in contrast to the closed-system device, allows the interchangeability between the scanned data file with the other interface. An intra-oral scanner could overcome some of the errors associated with traditional impression taking and cast production, as the digital output data can be fed directly into a digital workflow.

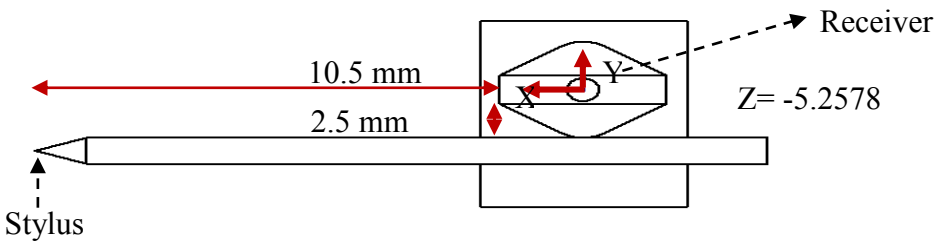


Figure 29: A schematic diagram of the magnetic sensor

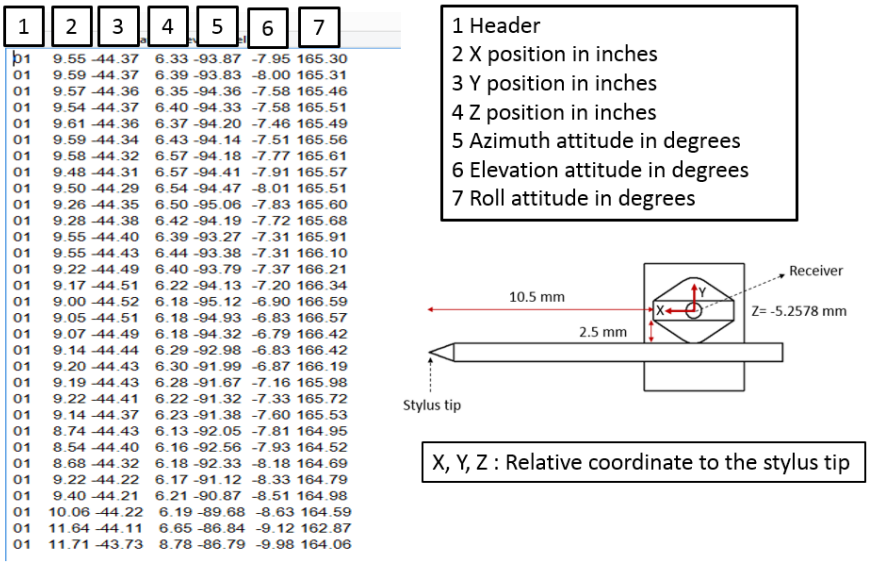


Figure 30: Example of the recorded magnetic sensor data (in inch)

For a 3D point cloud data, the data acquisition are performed using the magnetic sensor, which is connected to the computer. A set of data points in terms of XYZ coordinates and radial coordinates along the desired contour line are recorded simultaneously during the scanning process, as shown in Figure 30. A different type of digitizer could be selected for this purpose and in this study, the magnetic sensor is used due to the unavailability of a specific 3D digitizer. A relative coordinate of the receiver to the stylus tip is taken for the analysis since the stylus tip is used to trace the desired outline, while the receiver recorded the XYZ positions throughout the process. Finally, in the 2D image type of data, the patient's teeth image could be obtained from an intraoral camera, or other type of image capturing device. The 2D image could also be used for a specific treatment planning in the lingual orthodontics or maybe could be also manipulated for other usage, like for the definition of human dental arch, for example. A brief example of this type of input file would be described in Chapter 7.

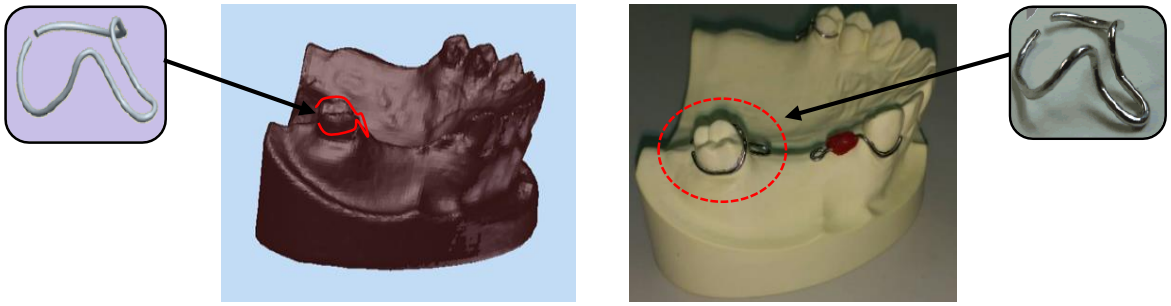
5.2 Bend points planning

In a CAD/CAM dental restoration, a computer is used to design the desired digital tooth restoration based on the patient teeth scan. The final design would be sent to the CNC milling machine as an input to control the movement of the CNC mechanism during the fabrication of the restoration part. Similarly, the same procedure could be used for the

CNC wire bending machine to design the desired wire shape. Alternatively, this could also be referred as the planning of the bending points. In this procedure, several cases have been considered, which produce different type of input file in terms of the STL file, a 3D point cloud data and the 2D teeth image. Therefore, a few methods have been explored in the bend points planning stage, in accordance with these input files. For this reason, different type of programs have been explored.

The automation of wire bending process through CNC technology requires the establishment of CAM data, most commonly known as G-code in the field of milling and turning process. The G-code controls the movement of the CNC machine in accordance with the desired manufacturing operation. In a wire bending, the preparation of CAM data requires information from the bend points planning stage, in terms of the XYZ Cartesian coordinates of the bend points. These coordinates would be processed through some mathematical formulae as introduced in the theory of 3D linear segmentation to establish bending parameters, with regard to the feed length (L), the rotation angle (β), and the bending angle (θ). Since the tool path generation theory in the standard G-code differs from the wire bending movement, therefore a B-code terminology is used in this study for the purpose of differentiating the approach. The following sub-sections discuss the bend points planning for each case study.

5.2.1 Case 1: Existence of physical data



(a) (b)

Figure 31: (a) bend points planning on the scanned dental cast (b) dental clasp fitting

In the first case, CAD software is used as the design platform to create the intended wire shape design. Figure 31 demonstrates the idea of this method. Since the output of this 3D digital scanner is in STL file format, therefore the file could be used in CAD

afterwards to create the desired wire shape. In this example, a dental wire clasp design has been created on the scanned model by a digital engineering approach. The target object is a dental impression created in a conventional dental impression technique, which could also be possible by a direct oral scanning using a handy digital scanner. After the desired target shape has been confirmed, the generation of the B-code through the developed B-code generation program would be executed, which translates the design information into the desired CAM data.

5.2.2 Case 2: Non-existence of physical data

A parametric fitting cscvn in Matlab is a periodic interpolating cubic spline curve, with a syntax of curve = cscvn (points). This syntax returns a parametric variational, or natural, cubic spline curve (in ppform) passing through the given sequence points (: j), j = 1: end. The parameter value t (j) for the j^{th} point is the accumulated square root of chord length, as indicated in Equation 9. If the first and last point coincide (and there are no other repeated points), then a periodic cubic spline curve is constructed. In this study, the cscvn interpolation program has been developed to fit the 3D data points according to the above theory.

$$\sum_{i<j} \sqrt{||points(:, i+1) - points(:, i)||_2} \tag{9}$$

The filtered point cloud data from the 3D filtering procedure would be further processed by the second program, which is the 3D point selection. In this procedure, an interface to select the desired 3D point is adopted from [63]. This program allows the user to select the desired 3D point cloud data by just clicking on the right point in 3D interface. The dataset could also be rotate, zoom and move to make the selection process more feasible. In this method, the selection of point is defined manually by the user and not automatically computed by the program. Therefore, the user must decide which points to be considered for the B-code generation. In this context, the expertise of user in deciding the appropriate point is required.

In this point selection program, the dataset file has to be initially imported into the program through Excel file, as illustrated by Figure 32a. Once the data has been imported, the program determines the total number of 3D point data in the file, as shown in Figure

32b. Simultaneously, the point selection interface would appear, as depicted in Figure 32c. At this point, the user can click on any desired point and once the point is selected, the index number of that point would be automatically printed in the command window, as indicated in Figure 32d. The selected point would be indicated by a red circle, as shown in Figure 32c. After the procedure is completed, the coordinates of each pre-selected point would be subsequently exported to Excel file. This file consists of all XYZ Cartesian coordinates of the pre-selected points and could be further employed in the following B-code generation program.

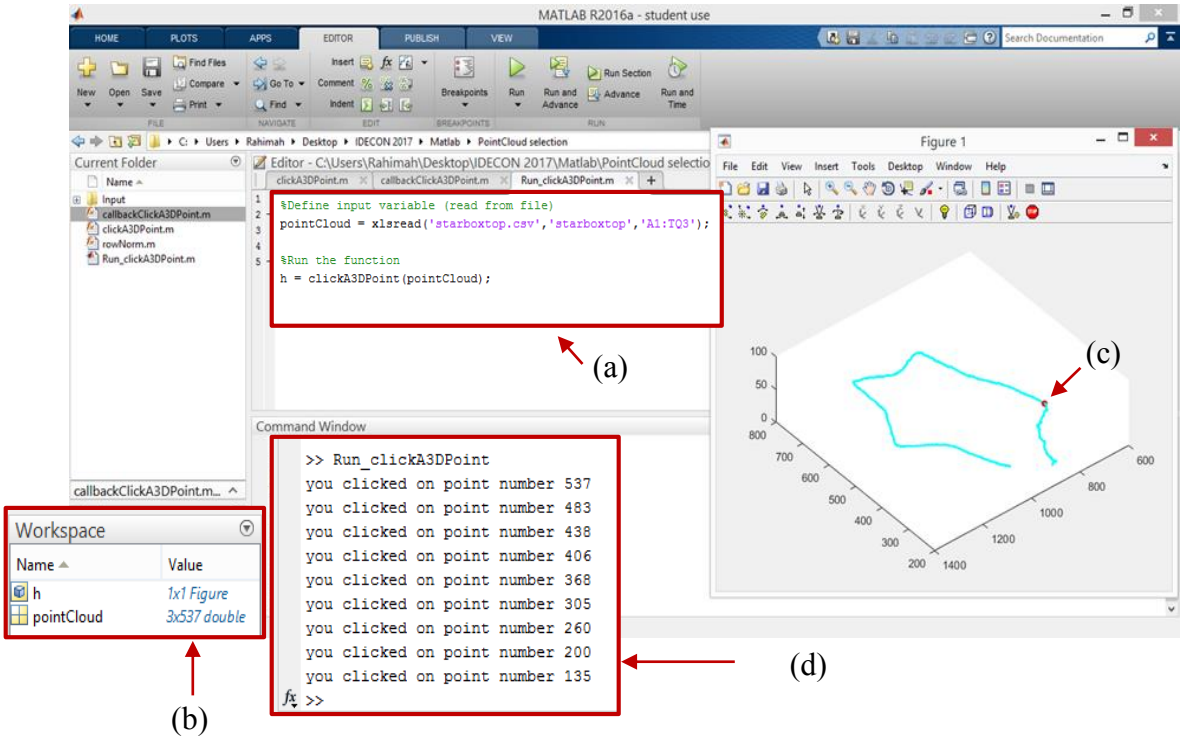


Figure 32: Interface for 3D point selection

5.2.3 Case 3: Employing 2D teeth image

In the third method, a 2D image is used as the input. This type of input data could be employed for the lingual arch-wire treatment planning, as reported in [22]. In this approach, the 2D image file is exported into the program, adopted from [64], where the user can estimate the desired treatment plan for the patient, as indicated in Figure 33. In this example, the treatment plan is made for the purpose of showing how the interface work, without any expertise about the real treatment plan. Then, the extraction of Cartesian coordinates (X, Y) of the pre-selected points into Excel could be executed by writing the following command, for instance.

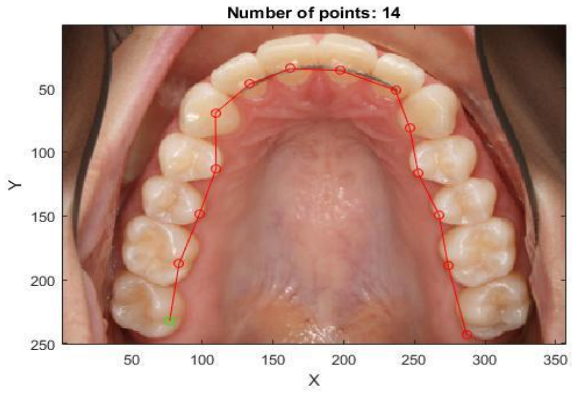


Figure 33: Interface for 2D point selection

In other word, the location of (X, Y) coordinates of each pixel representing landmarks of interest are highlighted and converted into a matrix of numerical data. These numerical data represent the location coordinates of all selected points. Subsequently, the XY data could be fed into the B-code generation program for the generation of B-code table in the fourth stage.

```
%Run the program%
Im1=imread('teeth.png');
[X0,Y0] = DS_getline(Im1,'plot');

%Write the output into file
xlswrite('xy.csv',X0,'A2:A20');
xlswrite('xy.csv',Y0,'B2:B20');
```

5.3 Algorithms for extracting parametric data from the IGES file

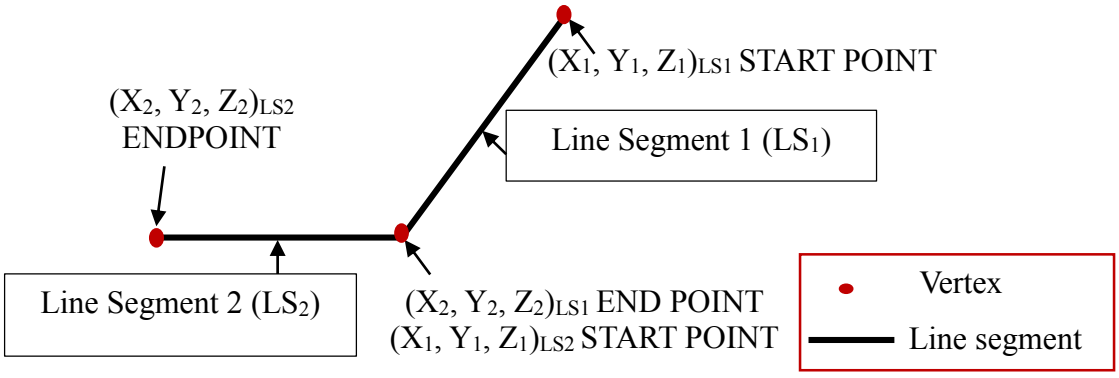


Figure 34: Line segment representation

According to Figure 25, the subsequent process after the bend points planning stage is the XYZ coordinate extraction. The XYZ coordinate for each point along the 3D line is important for the generation of B-code. In this process, the extraction procedure differs according to the interface used during the bend points planning stage. In the first case (STL file), for example, an automatic coordinate extraction program from the IGES file has been developed. IGES file format has been chosen as a standard format for the data transfer in CAD/CAM system. This .igs file gives a text file (ASCII format) where the coordinate of all points could be read according with the file formatting style. In contrast, the XYZ table would be automatically generated for the second case involving 3D point cloud data and the third case, which employs the 2D image, right after the bend points planning procedure is accomplished by the user, as discussed in the previous section.

Table 4: Structure of parameter data (P) section in the IGES file

Column	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***...***	***73-80***
Row 1	110	X_1	Y_1	Z_1	X_2	Y_2	Z_2		Sequence number
Row 2	110	X_1	Y_1	Z_1	X_2	Y_2	Z_2		Sequence number

Table 5: IGES file characterization and definition

Section	***ID letter***	***Description***
Start section	S	
Global section	G	
Direct entry section	D	All geometric entities are given here.
Parameter data section	P	All geometric entities are given here.
Terminate section	T	

IGES stores drawing information in an ASCII or binary neutral format which can then be exchanged between various users. In the ASCII 80 character format, the IGES file consists of a sequence of rows, each of 80 ASCII character position in a row is referred to as “column”, as illustrated in Table 4. The section identification letter occupies column 73 of each row and each section is distinguished by different characters (S, G, D, P, T), as shown in Table 5. Columns 73 to 80 are used to specify the sequence number of each row, while other data occupy columns 1 to 72. Based on this IGES file structure, all geometric information are given in the direct entry section (D) and the parameter data (P) section. D section organizes and gives structure to the information in the (P) section. There can be only one directory entry for each (P) section entity, as the (D) section entries differ

for various entities. Similar to most CAD systems, IGES is based on the concept of entities. Entities in IGES are divided into three categories which are geometric entities, annotation entities and structure entities. Geometric entities contain information that defines the object, such as points, lines and arcs. All of these geometries have their own entity index number, for example, Line = Type 110. In this recent work, Line entity is used for the generation of wire design in accordance to the 3D linear segmentation approach. Therefore, the information that is contained within this entity index in the IGES file line would be read by the program and extracted accordingly.

The developed coordinate extraction algorithm in this study directly reads the parameter data (P) section in the IGES file since it contains all the Cartesian coordinates of vertices along the 3D designed wire in a specific arrangement as shown in Table 6. In this line entity (Type 110) structure, the entity is defined in the parameter data section (P) using an end point and a start point, where the end point of a line segment 1 is the start point of a line segment 2 due to continuity, as illustrated in Figure 34. Therefore, this information is used once in the calculation since the B-code generation program requires only the Cartesian coordinates of each point along the 3D line segments. The developed IGES extraction program extracts and simplifies the data accordingly. As a result, the B-code table which comprises of all the desired parameters (L, β, θ) is subsequently generated and exported into Excel. A brief overview concerning to the IGES extraction would be briefly demonstrated by using an example.

In this work, the CAD design could be executed using commercially available CAD software package. The program has been written to read the IGES file and extract the coordinate information from the parameter data section using Matlab. This extractor is a code written in MATLAB that extracts the useful part of the IGES file, in such a way that only the basic information of the line segments is taken. Thus, other sections of the IGES file are ignored; in this case it refers to (S, G, D, and T). For more information about what these sections are and, in general, how an IGES file is organized, we refer the reader to [65].

Table 6 describes the algorithm which is designed for the IGES feature extraction program. This program imports numeric data from a text file as column vectors and reads data from the selected text file, defined in the first step. In this program, the user needs to define the start row and the end row of the desired location, for example; [xyz] = importIGES (FILENAME, STARTROW, ENDROW). The program will extract the data

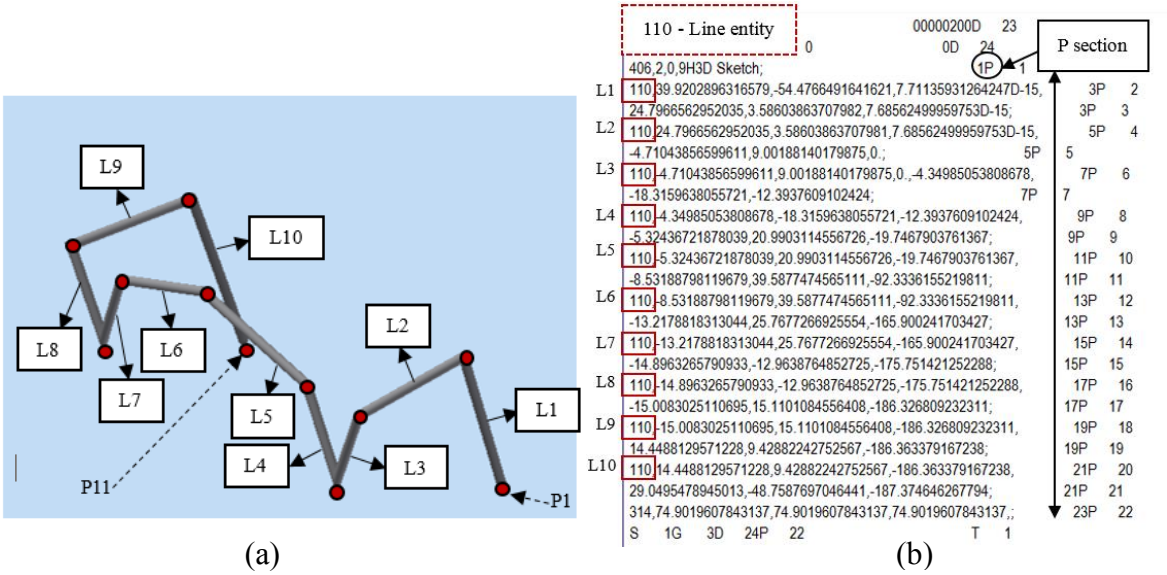
at the pre-defined location and read the parameter data from row to row. For example, if the user write [xyz] = importIGES ('example.igs', 30, 49), the program will read and extract the desired data in the 'example.igs' file from row 30 until row 49. Once the desired IGES file is called in Matlab, the program will read from row to row and extract the coordinates of vertices at each row as explained in 'Step 2'. At the same time, the program will also check for any irregularities in each row, in case there is any unnecessary information, such as alphabetical information etc. In the end, the program is closed and this generates a 'XYZ table' file simultaneously in Excel, saved at once in the same program folder.

Table 6: Algorithm for #importIGES program

Step	Instruction
Step 1	Open the IGES file fileID = fopen(filename,'r');
Step 2	Read columns of data according to format string. Row and column are counted by comparing identification character of the given 5 sections (S, G, D, P, and T) of the IGES file. Check every row to find the required entity (P). If found, the pointer will enter the parameter (P) section to extract the required entity index. If pointer find entity index 110, then the pointer extracts the required data from the parametric (P) section of the same entity as per its logic and store in output file named as 'XYZ table'. Once the data has been extracted, search for another P entity. Repeat step 2.
Step 3	Check for irregularities (if any).
Step 4	Remove multiple points from the original data. Consider only 4 decimals level.
Step 5	After completing checking for all rows, close the text file. fclose(fileID)

In the following example, the target shape generation in CAD is constructed based on the multiple 3D linear segments, as depicted in Figure 35a. This file is saved in IGES format, giving the ASCII format as shown in Figure 35b. Based on this IGES file, an automatic coordinate's extraction program has been developed and reported in [12], which reads the parametric data (P) section, which listed all the start point and the end point coordinate for each 3D line segment. The line geometry is numbered as 110 in this text file format. Therefore, based on this IGES file, the coordinate for P1-P11 is extracted automatically from row to row along the parameter data (P) section, as summarized in Table 7. This XYZ table is consecutively used in the B-code generation program which produces the B-code table, which would be further described in Section 6.2 for the 3D

simulation. In this simulation, the role of each parameter in controlling the movement of the CNC wire bender will be demonstrated. Thus, an introduction to the concept of a CNC dental wire bender in this study in relation to the B-code would be demonstrated.



(a) (b)

Figure 35: (a) The constructed model and, (b) the IGES file representing the design

Table 7: XYZ table

	x	*y*	*z*
P1	39.92	-54.48	7.71
P2	24.80	3.586	7.69
P3	-4.71	9.002	0.00
P4	-4.35	-18.32	-12.39
P5	-5.32	20.99	-19.75
P6	-8.53	39.59	-92.33
P7	-13.22	25.77	-165.9
P8	-14.90	-12.96	-175.75
P9	-15.01	15.11	-186.33
P10	14.50	9.43	-186.36
P11	29.05	-48.76	-187.38

5.4 Algorithms for B-code generation program

In the initial stage of this study, the B-code generation program was developed in Excel through some mathematical formulae as previously explained (Equation 1-Equation 8). Then, a limitation in terms of indicating the rotation sign for the rotational angle parameter has been encountered. In this context, the value obtained for each bending point could not exactly indicate whether the rotating mechanism shall turn clockwise or

counterclockwise which led to the second development of the converter in Matlab in order to resolve this issue. However, both converters have been used simultaneously in this study to check the calculated values for errors. In the end, aside from the limitation of indicating the sign of rotation, the Excel converter works adequately satisfying, in addition to the Matlab converter.

Shortly, the formulae used for both converters would be briefly discussed. Firstly, for the calculation of bending length (L) parameter, Equation 7 is used to compute the data for the converter in Excel. In this formula, x, y and z refer to the coordinate of a start point and an end point for each line segment, where root sum square is adopted in calculating the length, or the magnitude of the position vector. In Matlab, the command to calculate the vector (V1, V2) and also the length of the 3D vector (V)L = Vlength (V) is given, instead. This formula produces the straight length of the 3D linear segment (L), or the magnitude of the position vector in accordance to the root sum square formula.

In the meantime, Equation 2 is utilized in calculating the rotation angle (β), for the converter in Excel. The formula calculates the vector perpendicular to the current working plane using the cross vector and this is written in the Matlab as; plane V (i,:) = cross (V1,V2). Then, the length of vector perpendicular to the current working plane, applicable for a condition if (i>1) is calculated as; (PV1)L = Vlength (planeV (i-1,:)) and (PV2)L = Vlength (planeV (i,:)). In this context, the rotation angle for the first bend is always zero ((i)=0). In addition to this, the direction of the rotation from the previous plane to the current working plane is considered in the Matlab converter. The negative sign indicates a counterclockwise (ccw) rotation, while the positive sign indicates the clockwise (cw) rotation. Therefore, a command is written in the program, which is s = sign (V1. * cross (planeV (i-1,:)), planeV (i,:))'). The rotation angle is finally calculated in Matlab converter by writing this command: β(i) = acos(dot(planeV(i-1,:),planeV(i,:))/(PV1l*PV2l)).

Lastly, Equation 3 to Equation 6 describe the mathematical formulae used to compute the bending angle (θ) in the Excel converter. In this calculation, firstly, the length of a vector in two subsequent line segments is calculated and the included angle (α) between these two vectors is determined through dot product of these vectors. Inverse cosine of Equation 5 produces an included angle, α which needs to be subtracted from 180° or π, as shown in Equation 6. In the Matlab converter, however, the formulae to calculate the angle between the 3D vectors (V1, V2) have been simplified as follows; θ(i) = pi - acos(dot(V1,V2) / ((V1)L *(V2)L). The conversion from radian to degrees could be

easily done in this interface and for this study, the unit used for the bend angle and the rotation angle is in degrees.

Table 8: Algorithm for #xyz2bcode program

Step	***Instruction***
Step 1	Calculate V1 and V2 vector. V1 refers to the start point of the line segment and V2 refers to the end point.
Step 2	Calculate the length of the 3D vector; $(V1)_L$ = Vlength (V1), $(V2)_L$ = Vlength (V2).
Step 3	Calculate the angle (θ) between those 3D vector (V1, V2). $\theta_{(i)}$ = pi - acos(dot(V1,V2) / ($(V1)_L$ *$(V2)_L$));
Step 4	Calculate the vector perpendicular to the current working plane, let say plane ABC using the cross vector. planeV(i,:) = cross(V1,V2);
Step 5	Calculate the length of vector perpendicular to the current working plane. This is applicable for a condition; if (i>1). $(PV1)_L$ = Vlength (planeV (i-1,:)); $(PV2)_L$ = Vlength (planeV (i,:));
Step 6	Calculate the plane of bend before the current working plane. In this case, the first bend do not have the previous working plane which results to zero value for β.
Step 7	Calculate the direction of rotation from the previous plane to the current working plane. In this step, counterclockwise direction is given as –ve sign, while clockwise direction is given as +ve sign. s=sign (V1.* cross (planeV (i-1, :), planeV (i, :))');
Step 8	Calculate the rotation angle, β. β(i) = acos(dot(planeV(i-1,:),planeV(i,:))/(PV1l*PV2l)); Rotation of the first bend is always zero. β(i) = 0;
Step 9	The value of rotation angle in step 8 is multiplied with the sign of the rotation in step 7. β(i) = acos(dot(planeV(i-1,:),planeV(i,:))/(PV1l*PV2l))*s(1);
Step 10	Calculate the length of the straight by subtracting the tangent of the bend angle. L(i) = $(V1)_L$ - r(i)*tan(θ(i)/2); L(i+1) = $(V2)_L$ - r(i)*tan(θ(i)/2);
Step 11	All the Lβθ values are automatically exported to Excel columns.

Table 8 describes the B-code code generation program. It requires some inputs, in terms of XYZ coordinate of each vertex along the 3D line segments and the bend radius, *r*. The bend radius refers to the radius of the bend pin in the bending mechanism. In short, [L,β,θ] = xyz2bcode(Xc,Yc,Zc,r) is written, which means the output of (L, β, θ) is a result of executing 'xyz2bcode' function, with X, Y, Z and *r* as inputs. The development of the code is done in reference to [66]. On the other hand, the *r* value is set to 0 for the calculation of point-to-point length (L) which is based on the theory of 3D linear segmentation. In case if a radius bending is considered, the *r* value could be set to the required bend radius for the calculation of the bend length, instead of the straight length.

5.5 CNC dentistry wire bending process

In this study, the final aim is to bend a wire conforming to the data retrieved from a computer, known as B-code. The input data for this simulation consists of those generated B-code (L, β, θ). Theoretically, the feeding length (L) feeds the wire through a feeding roll towards the bending mechanism. The wire segment (L) is advanced by a feeding mechanism to a wire bending station. As the forward end of the severed segment enters the bending station, a stepping motor receives signals from a control system, which includes the microprocessor and computer, for turning the rotating mechanism (β) in either clockwise our counter clockwise and incrementally displacing a bend on the desired bend point in a controlled manner to produce a wire segment having the desired bending angle θ.

5.5.1 CNC dentistry wire bender

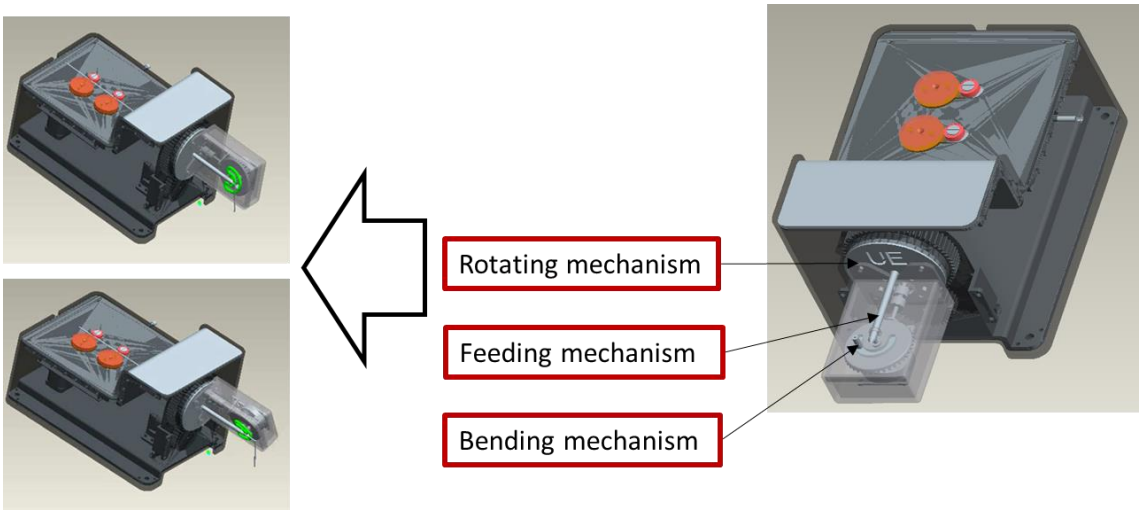


Figure 36: CNC dentistry wire bender and its main mechanisms

Three bending parameters have been recognized to control the CNC dentistry wire bender, known as the feed length (L), the rotation angle (β) and the bending angle (θ). The connection between these bending parameters and the wire bender are straightforward where the CNC wire bender is constructed from three mechanisms, in terms of the wire feeding mechanism, the rotating mechanism and lastly, the bending mechanism. The working principle of this wire bender is uncomplicated where with regard to this B-code (L, β, θ); L controls the feed of a straight length or the feed length of wire in mm, (β)

instructs the rotating mechanism to make a significant rotation (in degrees) either clockwise or counterclockwise in relation to the sign of (β) and finally, (θ) determines the corresponding bending angle for each bending operation. The rotation (β) of counterclockwise is considered as negative (-ve), while the clockwise rotation is alternatively represented as positive (+ve). The rotation angle (β) indicates the change of plane from the previous bend in accordance with the subsequent bending and the rotating mechanism has to adjust to this changes in preparation for the bending operation.

The model used for this demonstration was designed by another project of the lab. Modern computer technology linked with servo-mechanical control offers an excellent method for controlling the three bending mechanisms. For that reason, the movement of each mechanism is controlled by the B-code through L, β, and θ, respectively. The definition of these parameters was conducted according to the need of 3D non-planar bending through an analysis of an orthodontics retainer in CAD, as discussed in Chapter 4. The mechanics of this developing CNC dental wire bender operates in this way; L indicates the desired feed length to the bending mechanism, β instructs the rotating mechanism to rotate in preparation for the change of plane in the subsequent bending (3D non-planar case) while θ controls the bending mechanism to create the desired bend angle in each bending operation.

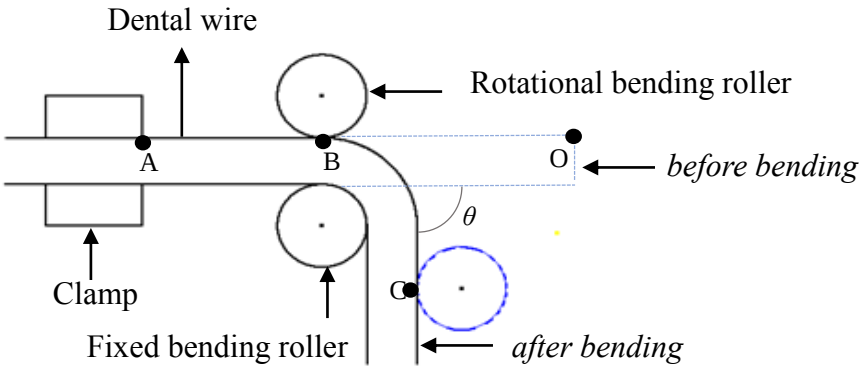


Figure 37: Graphical illustration of wire bending process

The working principle of this bending machine is relatively simple. At first, the straight dental wire is held in position by some set of rollers in accordance with the total value of L. In this wire bending process, the wire is pre-cut into the proper length which is estimated through the summation of the unbend length for all line segments, known as L, which refers to the point-to-point length in the line segment. Then, the value of this

unbend length is fed to the system by the feeding mechanism according to each line segment. Secondly, the rotating mechanism which is attached to the bending mechanism rotates according to the value of β. The rotation angle (β) indicates that the following bend plane differs from the current working bend plane, which starts from the second bend onwards. Therefore, the value of β in the first bend is always zero. The rotation of this mechanism is classified into two, which are counterclockwise and also clockwise position. In the B-code generation program, the notation of counterclockwise is defined as negative, while the positive sign is given to the clockwise rotation, respectively. Once the desired plane for the subsequent bending has been setup, finally the bending mechanism performs the desired bending operation based on the value of the feed length, L and bending angle, θ. In case if the value of bend radius is set to zero, then the point-to-point length would be similar to the feed length. As previously discussed, a theory of 3D linear segmentation considers a point-to-point length, rather than a radius bending. However, the B-code generation program could also estimate the required feed length in case the radius bending is required, which in this case we refer the point-to-point length (r=0) as L_{before} and the feed length in a radius bending (r=n) as L_{after} for differentiation. According to Figure 37, ABO refers to the length before the bending operation starts and ABC indicates the formed length after the bending operation ends, which refers to the radius bending.

5.5.2 Spring back consideration during bending process

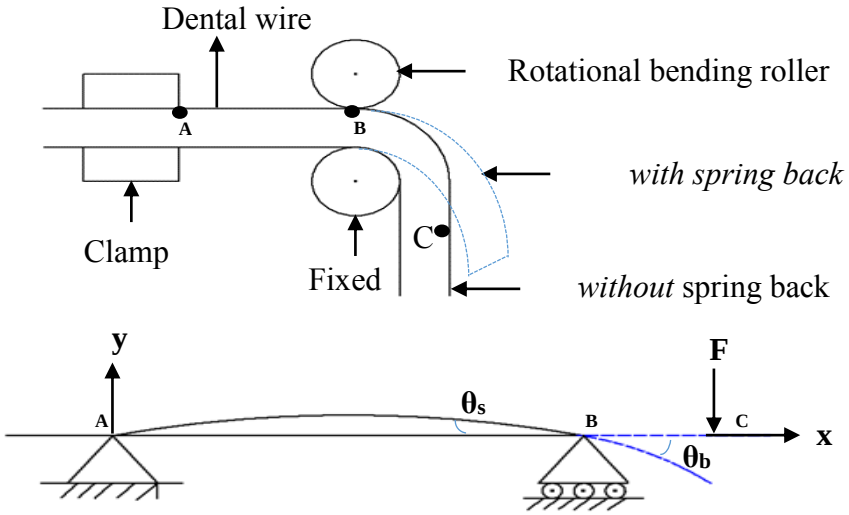


Figure 38: Graphical illustration of spring back effect

Recent advances in orthodontics wire alloys have resulted in a varied array of wires that exhibit a wide spectrum of properties. Several characteristics of orthodontic wires, for example are considered desirable for optimum performance during treatment, including spring back. Spring back is related to the ratio of yield strength to the modulus of elasticity of the material, which is also a measure of how far a wire can be deflected without causing permanent deformation. During the wire bending operation, the spring back causes the formed angle to differ from the calculated bending angle, θ. This happens when the material tries to return to its original position after being bent. Therefore, different material properties lead to different spring back effects and in orthodontics, there are different type of wires, such as nickel titanium (NiTi) wire, titanium molybdenum (TMA) wire, cobalt-chromium wire and beta-titanium. The spring back calculation for each material might vary, in accordance to the material properties. According to [67], the spring back tendency of stainless steel and cobalt-chromium is low, while it is high and average for nickel titanium and beta-titanium, respectively.

The spring back effect of the wire is graphically illustrated as in Figure 41. During the bending process, the wire is fixed at one end with a clamp to prevent rotation and movement. On the other hand, the rotational bending pin moves clockwise around the gyration centre of the fixed mould to achieve the desired bend shape by applying the bend force to the wire. After the rotational bending pin rotates counter clockwise to its original position and the load is removed, the spring back is generated due to elastic deformation. The idea of illustrating the simplified model of elastic bending phase as in is adopted from [68], which reported that the process of wire bending consists of two parts, AB and BC, respectively. The BC part forms the wire, which is the elastic-plastic deformation state while the AB part is affected by the bending moment of the BC part, generating the bending deformation and is in an elastic deformation state. In Figure 41, θ_b is the generated bending angle during the operation and θ_s is the spring back angle. According to [69], in elastic deformation stage, although the spring back angle itself is small, the proportion of it in bend angle is very large. In plastic deformation stage, although the spring back angle itself is large, the proportion of it in bend angle is relatively small. The compensation of this spring back angle is necessary in order to produce an accurate bend [70] and this is confirmed by [51] which has shown the reduction of the maximum and minimum error ratios of the formed arch-wire before and after considering the spring back phenomenon into the calculation.

During the bending operation, the wire is compressed and stretched through the bend [71]. In tube bending, the tensile area always deal with buckling effect and the compressive area is always affected by wrinkling effect [70]. In dental wire bending literature, no such effect has been considered by other researchers. Therefore, it will not be discussed in the present study. Arch-wire in orthodontics, for example is a kind of metal material with high elasticity. According to [72], spring back is a complex phenomenon related to actual bending force, angle and material properties. Relevant researches on spring back effect fall into two aspects, which are the theoretical calculation and also the finite element analysis (FEA). The author concluded that the derived theoretical formula could differ from case to case and it is hard to construct a general formula for all bending configurations in practice. In fact, computing deformation in FEA software heavily relies on obtaining the material properties of dental wire from manufacturer. Therefore, the present study adopted the over bend prediction by [72] in considering the spring back phenomenon in the future. The spring back factor, K_s is defined as:

$$\mathrm{Ks} = \left(\frac{\theta_b}{\theta_f}\right) \tag{10}$$

In this case, θb is the calculated bending angle and θ_f is the formed angle due to spring back effect. The spring back factor ranges from 0 to 1, and is related to material properties, bending angle, θ_b and also the bend radius, r. This author recommends the over bend of the wire to be carried out in accordance with the value of the formed angle, θ_f. In order to proceed, an over bend allowance, $\Delta\theta$ can be calculated based on Equation 11.

$$\Delta\theta = \theta_b - \theta_f = \frac{1}{\mathrm{Ks}-1} \times \theta_f \tag{11}$$

The equation is generalized and for any given target angle, θ, the over bend allowance for pre-compensation can be expressed by:

$$\Delta\theta\,(\theta) = \frac{1}{\mathrm{Ks}-1} \times \theta \tag{12}$$

θ is the desired target angle and $\Delta\theta$ is the predicted over bend allowance. However, in order to use this method for the spring back consideration, a Finite Element Analysis (FEA) analysis or the actual bending experiment has to be pre-conducted for different type of dental wire. This is executed in order to define the value of spring back factor, K_s. In addition, the material properties for each wire has to be initially requested from the manufacturer since Australian stainless steel arch wire might be slightly different from the China stainless steel arch-wire, for instance.

For these reasons, the spring back analysis is not being considered in the present work. Firstly, the actual wire bending experiment is not feasible due to the wire bending system which is not established yet. On the other hand, the FEA analysis requires the material properties of each different type of wire. This is another challenge since the manufacturer has to be contacted in order to acquire this information. As discussed by many researchers, the material properties of dentistry wires vary according to the manufacturer and are not standardized as the industrial wires. In this context, the stainless steel wire, for example, which comes from a manufacturer from China might have a slightly different material properties from the one which is produced by Australia. This has led to the difficulty in constructing a standard spring back algorithm in the field of dentistry applications.

CHAPTER 6

IMPLEMENTATION OF B-CODE

In this section, the implementation of B-code is demonstrated through a case study. The present work is designed to test the developed vertices coordinate extraction program from the IGES file, in addition to the B-code generation program at once. These two main procedures could be consecutively executed in the same environment. Several functions and commands developed by the author to extract the coordinates of vertices and employing those data for the CNC bending code generation have been introduced in Section 5.3 and Section 5.4, respectively. Therefore, case 1 type of input data has been chosen for this demonstration.

6.1 B-code validation

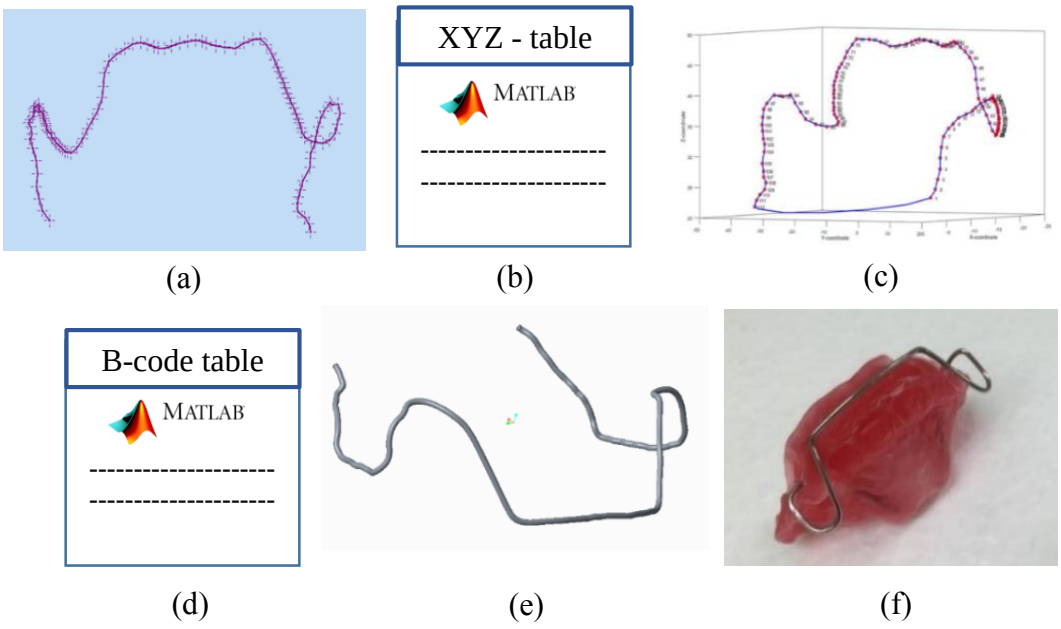


Figure 39: Required procedure to prepare the wire bending (a) bend points planning (b) XYZ extraction (c) 3D interpolation (d) B-code generation (e) 3D wire bending (f) fitting

In this validation work, the bend points planning of the STL file is initially performed through the concept of multiple 3D linear segments. After the bend points

planning is completed, the output of the shape design is saved in the IGES file format. The IGES file (*.igs) is subsequently used by the IGES extraction program to extract all the coordinates of vertices as shown in Figure 39(b) and Figure 40(a), producing the XYZ table. After that, additional interpolations of the coordinates are performed in Matlab to visualize the output and to confirm that the program has extracted the right points, as shown in Figure 39(c). Then, the XYZ table is fed into the B-code generation program to produce the bending parameters (L, β, θ), as illustrated in Figure 39(d) and Figure 40(b). The B-code is important to produce the desired dental wire shape, as simulated in Figure 39(e). Figure 39(f) depicts the example of the wire in the real application. In this example, the bend radius, r used is 0.4 mm. The actual results for both XYZ table and the B-code table for are depicted in Figure 40(a) and Figure 40(b), respectively.

In order to demonstrate the 3D simulation of the B-code in the following section, the present example is not appropriate due to a large number of bending (n^{th} = 110). In this B-code validation, the digital shape generation involved with a large number of 3D linear segments, which are 111. The reason for experimenting with a large number of line segments is to test the robustness of both programs. For instance, the IGES-XYZ extraction program has extracted all the XYZ coordinates of 112 vertices accurately. The generated XYZ table has been compared and checked manually with the 3D wire model in CAD which verified the result. Then, the XYZ-table is used for the B-code generation program at a later stage. In this analysis, the purpose is to validate the developed B-code generation program and for the same reason, a large number of XYZ coordinates are used from the previous program. In addition, difference values of bend radius, r have been tested during this analysis. The reason for experimenting with these r values is to show that the B-code generation program is capable to indicate whether the actual bending is possible or not by showing the negative bend length, L_{after} in the table if the input value of r is too large for the line segment. In the actual 3D wire bending machine, the bend radius, r refers to the radius of the bending rollers. Therefore, the user could predict whether the actual wire bending is possible or not by giving the bend radius, r of the machine to the program. In this demonstration, two different r values are used, which are 1 mm and 0.4 mm. The use of 1 mm bending radius has indicated a negative bend length, while the other 0.4 mm bending radius is theoretically possible to realize the actual bending of the designed shape.

XYZ - table

MATLAB

B-code table

MATLAB

VERTEX	X	Y	Z				
V1	-9.04637	8.757101	23.42624	V57	-24.0083	-10.7899	47.33425
V2	-9.22815	9.961924	24.85802	V58	-23.9046	-11.9696	47.65004
V3	-9.32235	10.55846	26.4714	V59	-23.5041	-13.4593	48.06282
V4	-9.54943	10.98252	28.17611	V60	-23.0729	-14.8194	48.28159
V5	-9.68098	10.79155	29.98848	V61	-22.8277	-16.1133	48.47095
V6	-9.76424	11.04422	31.74148	V62	-22.7662	-17.1757	48.31418
V7	-9.84143	11.61964	33.50087	V63	-22.4751	-17.9064	48.05539
V8	-10.5413	12.10171	34.7442	V64	-22.4432	-19.5161	47.68014
V9	-11.7188	11.74789	35.45847	V65	-22.0393	-21.109	47.47735
V10	-12.7167	11.93849	36.41398	V66	-21.2679	-22.8583	47.33405
V11	-13.999	12.33033	36.93062	V67	-20.2918	-24.3467	48.08133
V12	-15.4068	12.14774	37.40608	V68	-18.9694	-25.8942	48.54548
V13	-16.3828	11.62628	37.83764	V69	-17.1934	-27.1435	48.65657
V14	-17.5213	11.35871	38.49419	V70	-16.9881	-28.1842	47.75381
V15	-18.5107	11.43742	38.83199	V71	-16.5343	-29.0422	46.59529
V16	-19.4706	11.3561	39.12865	V72	-16.3906	-29.5339	45.6787
V17	-20.2027	11.03021	39.29815	V73	-16.0291	-30.0211	44.64266
V18	-20.5078	10.60282	39.57018	V74	-15.7355	-30.2788	44.05486
V19	-20.8006	10.52009	39.05875	V75	-15.5091	-30.7143	43.20573
V20	-20.9647	10.49578	38.7131	V76	-15.606	-31.0632	42.53236
V21	-21.2216	10.25073	38.63183	V77	-15.5036	-31.3717	41.80031
V22	-21.4921	9.981608	38.38222	V78	-15.4955	-31.6763	41.06707
V23	-21.7873	9.838516	38.01243	V79	-15.8128	-32.058	40.3714
V24	-22.0313	9.69422	37.58695	V80	-15.9386	-32.3799	39.70518
V25	-22.2064	9.584748	37.25712	V81	-16.2589	-32.7937	39.06316
V26	-22.3664	9.390005	36.91139	V82	-16.4145	-33.1811	38.20062
V27	-22.4516	9.301899	36.59515	V83	-16.7269	-33.5073	37.49257
V28	-22.6481	9.187003	36.01251	V84	-16.8457	-33.7615	36.94053
V29	-22.899	8.931175	35.68854	V85	-17.1722	-34.0397	36.24015
V30	-23.0861	8.573284	35.51636	V86	-17.7583	-34.2498	35.66657
V31	-23.2789	8.368414	34.96972	V87	-18.4308	-34.7227	34.96972
V32	-23.4134	8.053987	34.85299	V88	-18.8276	-35.7331	34.47935
V33	-23.4949	7.834605	34.52491	V89	-18.0085	-37.7263	34.18415
V34	-23.6015	7.596228	34.15138	V90	-16.5361	-38.9171	34.63722
V35	-23.7633	7.219299	33.89378	V91	-15.0392	-39.8071	35.53389
V36	-23.9068	6.874577	33.6569	V92	-14.2239	-40.145	36.78233
V37	-24.0565	6.533048	33.38551	V93	-13.4231	-40.6697	38.3032
V38	-24.2349	6.086929	33.05679	V94	-12.3353	-40.3605	39.62606
V39	-24.4344	5.669985	32.96184	V95	-10.0972	-39.2708	39.49285
V40	-24.6391	5.0899	32.82661	V96	-8.08858	-38.9821	39.81022
V41	-24.7	4.018128	33.61291	V97	-5.93121	-37.9338	38.63183
V42	-24.7137	3.09839	34.84682	V98	-4.94783	-37.4242	37.71679
V43	-24.6019	2.314283	36.12377	V99	-4.41137	-36.917	36.43451
V44	-24.4418	1.38599	37.53409	V100	-4.09352	-36.6285	35.19941
V45	-24.1963	0.496942	38.8407	V101	-4.06601	-36.4255	34.20601
V46	-24.3794	-0.16979	40.51991	V102	-3.7643	-35.7431	32.94483
V47	-24.2139	-0.39484	42.01336	V103	-3.90299	-35.5721	31.99937
V48	-24.1227	-0.56524	43.88002	V104	-3.82725	-35.5348	30.82723
V49	-23.5787	-1.34062	45.54862	V105	-2.63105	-34.3155	28.9613
V50	-23.4865	-2.6367	46.52865	V106	-2.62246	-33.9113	27.71998
V51	-23.4435	-4.11579	47.2779	V107	-2.478	-33.6428	26.74424
V52	-23.4556	-5.396	47.93618	V108	-2.79956	-33.3819	25.83571
V53	-23.8081	-6.80429	48.12644	V109	-2.58221	-33.4878	24.71581
V54	-23.9416	-7.98359	47.85663	V110	-2.47908	-34.733	23.88217
V55	-23.8866	-9.03101	47.55822	V111	-2.1729	-35.3767	22.9701
V56	-23.9543	-9.78611	47.3479	V112	-2.01468	-35.7018	21.87935

	L_{before}	L_{after}	β	θ
Bend1	1.88	1.81	0.00	19.88
Bend2	1.72	1.63	41.82	7.58
Bend3	1.77	1.67	-41.72	20.14
Bend4	1.83	1.71	-164.51	14.26
Bend5	1.77	1.69	-4.65	9.90
Bend6	1.85	1.73	-85.62	25.47
Bend7	1.51	1.25	-43.75	43.96
Bend8	1.42	1.17	167.87	25.49
Bend9	1.39	1.22	-96.69	22.76
Bend10	1.44	1.27	-99.62	23.06
Bend11	1.50	1.35	-22.06	19.67
Bend12	1.19	1.06	-141.74	15.64
Bend13	1.34	1.22	-56.38	19.30
Bend14	1.05	0.95	-134.48	9.09
Bend15	1.01	0.91	0.94	19.24
Bend16	0.82	0.63	-50.80	32.45
Bend17	0.59	0.06	131.01	92.35
Bend18	0.60	0.16	-2.72	6.17
Bend19	0.38	0.15	-175.38	56.92
Bend20	0.36	0.08	159.77	20.33
Bend21	0.46	0.31	40.78	20.83
Bend22	0.49	0.39	-68.21	8.41
Bend23	0.51	0.48	-2.70	1.79
Bend24	0.39	0.34	-77.44	11.12
Bend25	0.43	0.34	125.23	15.06
Bend26	0.34	0.27	73.62	5.65
Bend27	0.63	0.51	106.00	27.74
Bend28	0.48	0.30	29.78	23.63
Bend29	0.44	0.21	158.43	41.06
Bend30	0.61	0.29	-175.35	46.06
Bend31	0.36	0.06	164.62	35.72
Bend32	0.40	0.27	78.67	2.21
Bend33	0.46	0.37	104.65	22.93
Bend34	0.48	0.40	114.89	0.60
Bend35	0.44	0.43	77.52	3.76
Bend36	0.46	0.44	-150.85	2.26
Bend37	0.58	0.49	-51.44	23.03
Bend38	0.47	0.37	106.11	6.04
Bend39	0.63	0.42	-80.05	50.96
Bend40	1.33	1.08	-15.66	17.16
Bend41	1.54	1.45	35.69	7.02
Bend42	1.50	1.47	104.54	2.13
Bend43	1.70	1.68	-43.17	3.52
Bend44	1.60	1.52	-153.33	19.26
Bend45	1.82	1.69	91.74	17.76
Bend46	1.52	1.44	-88.33	4.82
Bend47	1.88	1.78	-170.32	23.72
Bend48	1.92	1.73	55.83	30.51
Bend49	1.63	1.48	-18.05	10.37
Bend50	1.66	1.62	90.06	1.99
Bend51	1.44	1.35	-64.32	23.54
Bend52	1.46	1.30	-51.82	21.63
Bend53	1.22	1.11	-49.50	9.66
Bend54	1.09	1.03	164.33	7.83
Bend55	0.79	0.71	94.12	14.86
Bend56	1.01	0.89	19.04	17.64
Bend57	1.23	1.13	60.68	9.68
Bend58	1.60	1.54	67.06	6.74
Bend59	1.44	1.40	107.42	6.80
Bend60	1.33	1.24	-60.77	18.14
Bend61	1.08	0.94	-83.57	20.41
Bend62	0.83	0.68	165.06	20.45
Bend63	1.65	1.53	138.92	14.25
Bend64	1.66	1.57	10.73	9.91
Bend65	1.92	1.78	-55.00	28.60
Bend66	1.93	1.79	123.95	12.09
Bend67	2.09	1.98	-22.50	17.32
Bend68	2.17	1.89	92.22	58.90
Bend69	1.39	1.11	-98.65	15.16
Bend70	1.51	1.41	83.64	12.56
Bend71	1.05	0.97	-123.99	10.10
Bend72	1.20	1.14	-6.23	7.19
Bend73	0.71	0.64	-174.70	11.71
Bend74	0.98	0.87	-15.13	20.62
Bend75	0.76	0.64	-163.37	15.31
Bend76	0.80	0.72	160.20	6.77
Bend77	0.79	0.69	17.23	23.18
Bend78	0.85	0.73	177.76	12.48
Bend79	0.75	0.66	-166.52	14.72
Bend80	0.83	0.72	-175.26	15.74
Bend81	0.96	0.86	151.72	12.50
Bend82	0.84	0.76	177.50	10.78
Bend83	0.62	0.54	167.07	12.71
Bend84	0.82	0.70	10.64	20.43
Bend85	0.85	0.73	115.54	11.76
Bend86	1.08	0.92	-2.84	32.19
Bend87	1.19	0.91	23.90	44.95
Bend88	2.18	1.88	-9.75	35.55
Bend89	1.95	1.76	-27.02	15.79
Bend90	1.96	1.80	-20.71	28.15
Bend91	1.53	1.41	-73.99	6.69
Bend92	1.80	1.67	152.06	29.55
Bend93	1.74	1.43	42.12	53.26
Bend94	2.49	2.22	132.31	21.38
Bend95	2.05	1.84	-148.94	39.06
Bend96	2.67	2.48	23.62	13.45
Bend97	1.44	1.31	-15.05	22.82
Bend98	1.48	1.36	35.26	10.77
Bend99	1.31	1.23	-39.60	12.61
Bend100	1.01	0.90	-129.57	19.66
Bend101	1.47	1.30	163.22	27.02
Bend102	0.97	0.82	102.29	14.58
Bend103	1.18	0.98	82.46	38.60
Bend104	2.53	2.28	153.78	31.38
Bend105	1.31	1.16	135.05	8.18
Bend106	1.02	0.90	-162.39	26.92
Bend107	1.00	0.77	144.28	36.47
Bend108	1.15	0.82	-58.28	50.79
Bend109	1.50	1.23	160.58	23.59
Bend110	1.16	1.00	46.83	19.67
	1.15	1.08		

(a) (b)

Figure 40: (a) XYZ table and, (b) B-code table

6.2 B-code demonstration through a case study

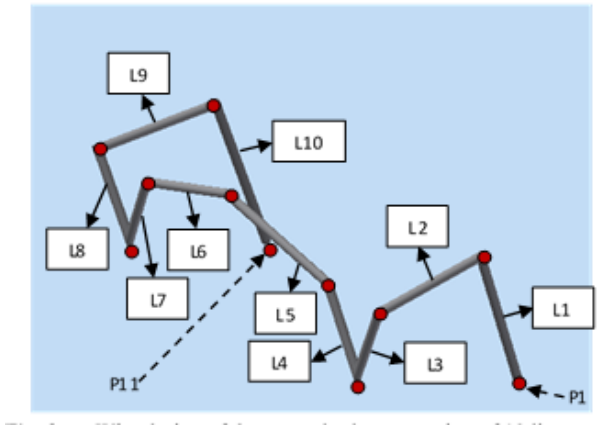
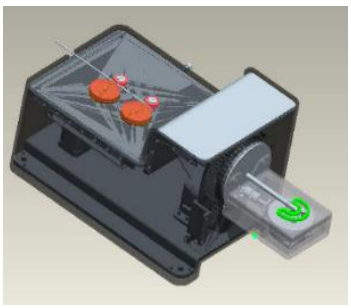


Figure 41: Target shape

Table 9: B-code table

BEND	*L (mm)*	*β (deg.)*	*θ (deg.)*
1	60	0.00	66
2	30	-19.83	94
3	30	-108.84	145
4	40	-178.73	65
5	75	3.61	25
6	75	-0.90	65
7	40	179.03	145
8	30	89.44	100
9	30	22.07	65
10	60		

In this section, the previous example as indicated in Section 5.3 would be further used for the demonstration of 3D wire bending process by using the introduced CNC dental wire bender concept. A simple target shape is used which has been simplified from the actual dental appliance for the purpose of demonstration, as depicted in Figure 41. Based on the wire design which is developed in CAD through a theory of 3D linear segmentation, the whole model is constructed from 10 3D linear segments. In total, 11 3D points are significant for the following generation of B-code. For that reason, the XYZ Cartesian coordinates of each point which has previously been extracted and summarized in Table 7 is employed by the B-code generation program to create the required bending parameters, as illustrated in Table 9. According to this B-code table, the number of bends required for this bending operation is 9. Each bending process would be briefly discussed by using the 3D wire bender concept, in relation to the wire bending parameters (L, β, θ).



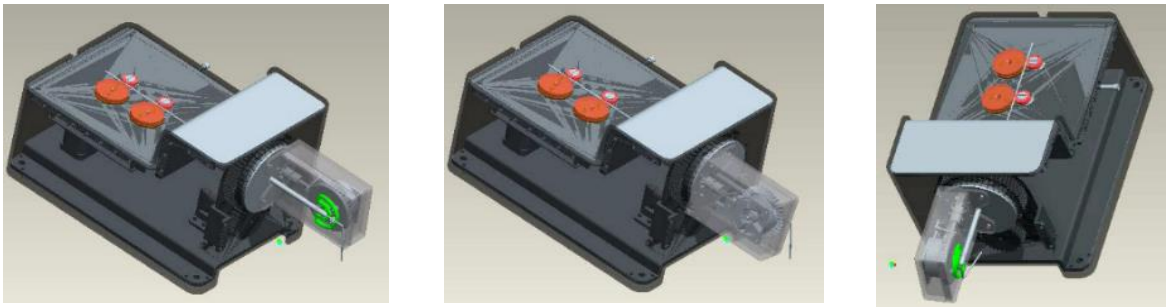
Start

L_1=60mm, β_1=0°

θ_1=66°

Figure 42: First bend

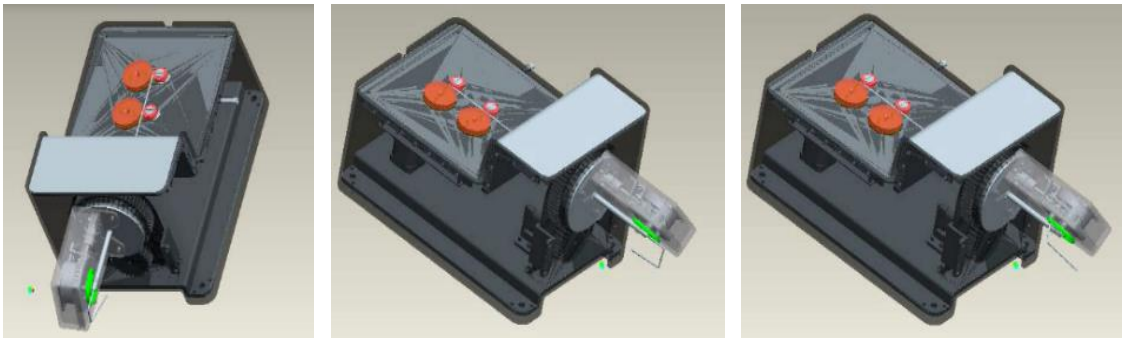
Figure 42-50 illustrates the bending sequence for all bends, starting from bend 1 until bend 9. Initially, the bending mechanism is in a normal position, as illustrated in Figure 42. Based on Table 9, the straight length (L) that needs to be feed by the sending mechanism is 60 mm. Therefore, in step 2, 60 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates and make the first bend of 66 degrees (θ). The rotation angle (β) for the first bend is zero.



$L_2 = 30\text{mm}$ $\beta_2 = -19.83°$ $\theta_2 = 94°$

Figure 43: Second bend

In the second bend, the bending mechanism is in the previous position. According to Table 9, the straight length (L) that needs to be feed by the sending mechanism is 30 mm. Therefore, 30 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 19.83 degrees (ccw) and creates the second bend of 94 degrees.



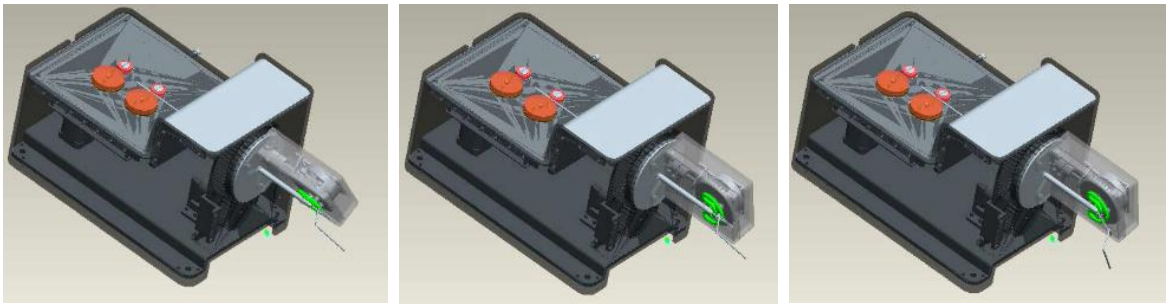
$L_3 = 30\text{mm}$ $\beta_3 = -108.84°$ $\theta_3 = 145°$

Figure 44: Third bend

In the third bend, 30 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 108.84 degrees (ccw)

and makes the third bend of 145 degrees, as shown in Figure 44.

Subsequently in the fourth bend, 40 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 178.73 degrees (ccw) and creates the second bend of 65 degrees.



L_4 = 40mm β_4 = -178.73° θ_4 = 65°

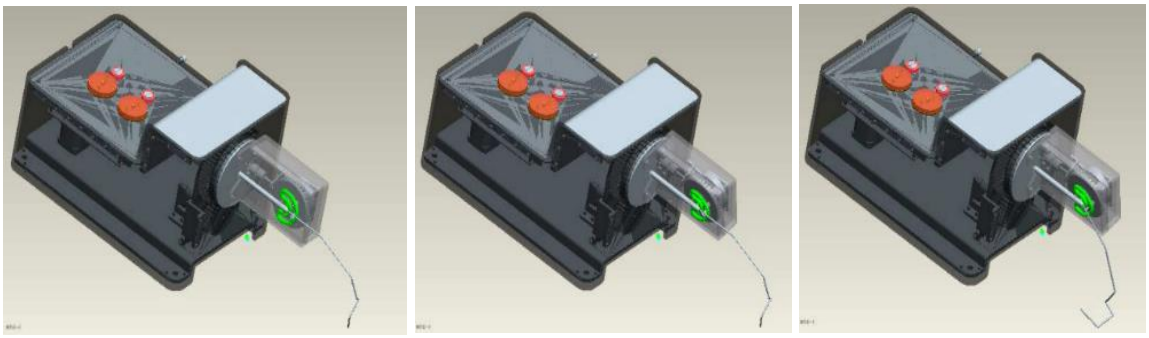
Figure 45: Fourth bend

In the fifth bend, 75 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 3.61 degrees (cw) and makes the fifth bend of 25 degrees, as indicated in Figure 46.

L_5=75mm β_5=3.61° θ_5=25°

Figure 46: Fifth bend

In the sixth bend, 75 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 0.90 degrees (cw) and creates the sixth bend of 65 degrees. The sequence of this bending process is demonstrated in Figure 47.



L_6 = 75mm β_6 = 0.90° θ_6 = 65°

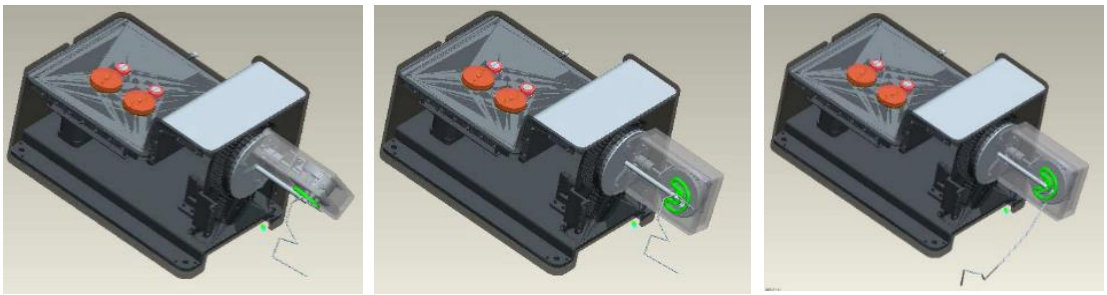
Figure 47: Sixth bend

In the seventh bend, 40 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 179 degrees (ccw) and makes the seventh bend of 145 degrees.

L_7=40mm β_7=-179.03° θ_7=145°

Figure 48: Seventh bend

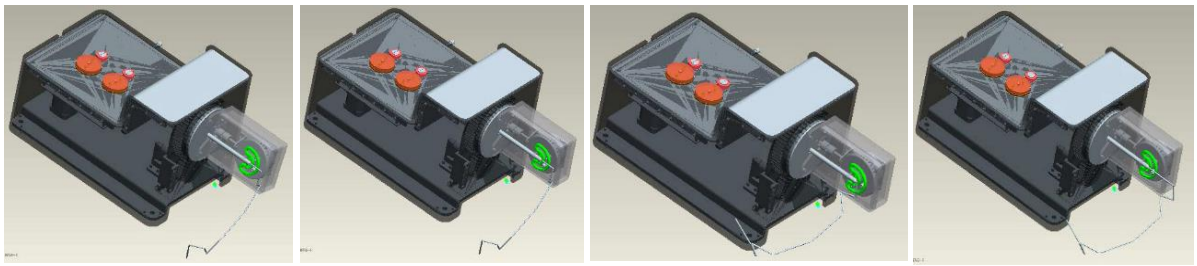
In the eighth bend, 30 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 89.44 degrees (ccw) and creates the eight bend of 100 degrees.



L_8 = 30mm β_8 = -89.44° θ_8 = 100°

Figure 49: Eighth bend

In the last bend, 30 mm of wire is fed by the wire sending mechanism towards the bending mechanism. Then, the turning mechanism rotates about 22 degrees (ccw) and makes the ninth bend of 65 degrees. Then, the final straight length of 60 mm is fed to end the wire bending process, which represents the length of the tenth 3D linear segment, as shown in Figure 41.



$L_9 = 30\text{mm}$ $\beta_9 = -22.07°$ $\theta_9 = 65°$ Finish

Figure 50: Ninth bend

The bending operation ended when the last feed of straight wire is forwarded by the sending mechanism. A target shape as indicated in Figure 41 is chosen for this demonstration for the purpose of demonstrating the general idea of the proposed methodology. As mentioned earlier, the process flow of this CNC dental wire bending starts with data scanning, follows by bend points planning and the extraction of XYZ Cartesian coordinates for each bend point. Subsequently, the Cartesian coordinates are processed by the B-code generation program through some mathematical formulae to produce the desired CAM data (B-code) to control the movement of the CNC dental wire bender. The simulation of this CNC wire bending operation in order to show how the B-code controls the desired 3D wire bending operation with regard to the desired target shape has successfully been demonstrated in this chapter.

6.3 Summary of B-code implementation analysis

To sum-up, the implementation of B-code has successfully been demonstrated through a case study, which tested the IGES-XYZ coordinate extraction program and the B-code generation program at once. Furthermore, 3D simulations of the selected target shape has also been elaborated which employed the concept of the 3D wire bender (feed, rotate, bend) which has been theoretically discussed in Chapter 2. This has proven the capability of the generated B-code to realize the desired 3D wire bending operation.

CHAPTER 7

3D SIMULATION OF B-CODE

In this chapter, 3D wire bending simulations of the generated B-code which is initially processed by the 3D point cloud data filtering and the 3D point cloud data selection programs are executed in order to check the reliability of the introduced programs. In addition, 3D wire bending simulations of the generated B-code from the 2D image input file has also been examined. This clarifies to the third objective of this study, which requires a validation of the B-code generation program.

7.1 Pre-processing of 3D point cloud data

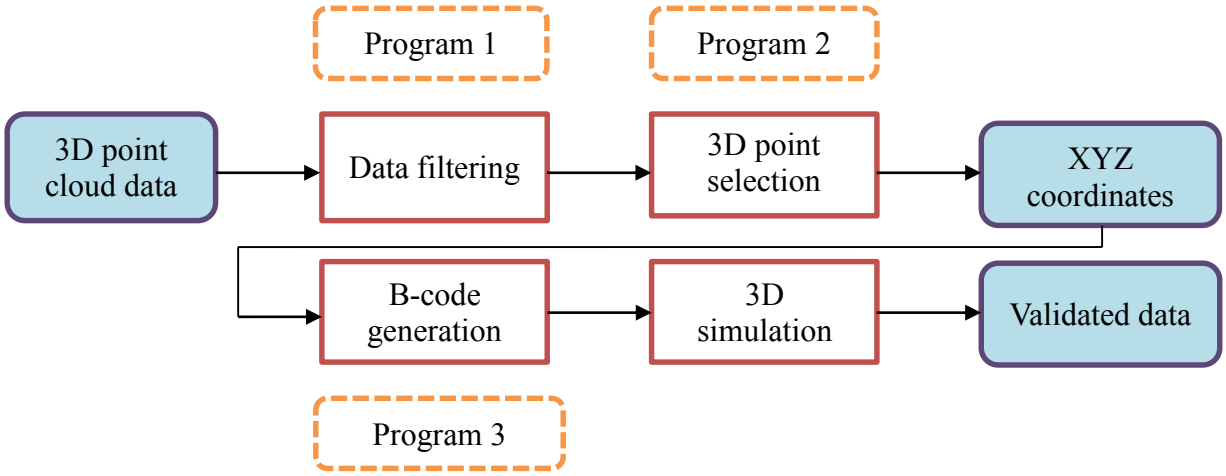
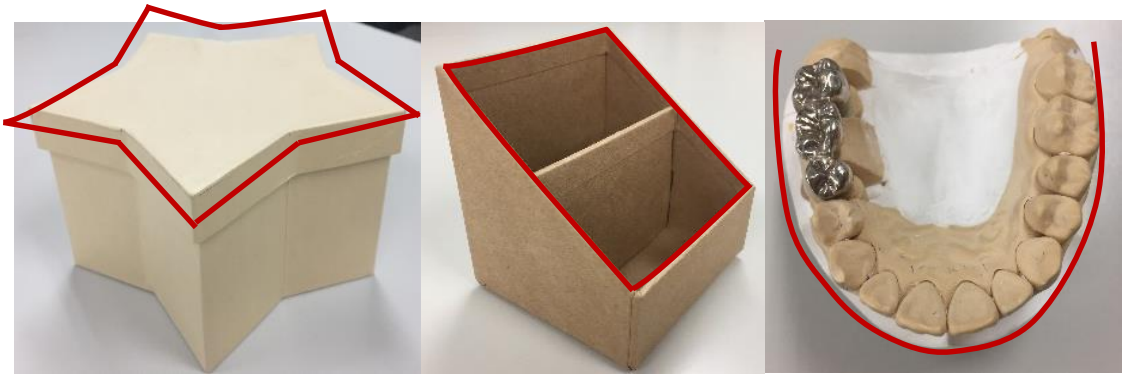


Figure 51: A brief methodology of 3D point cloud data processing

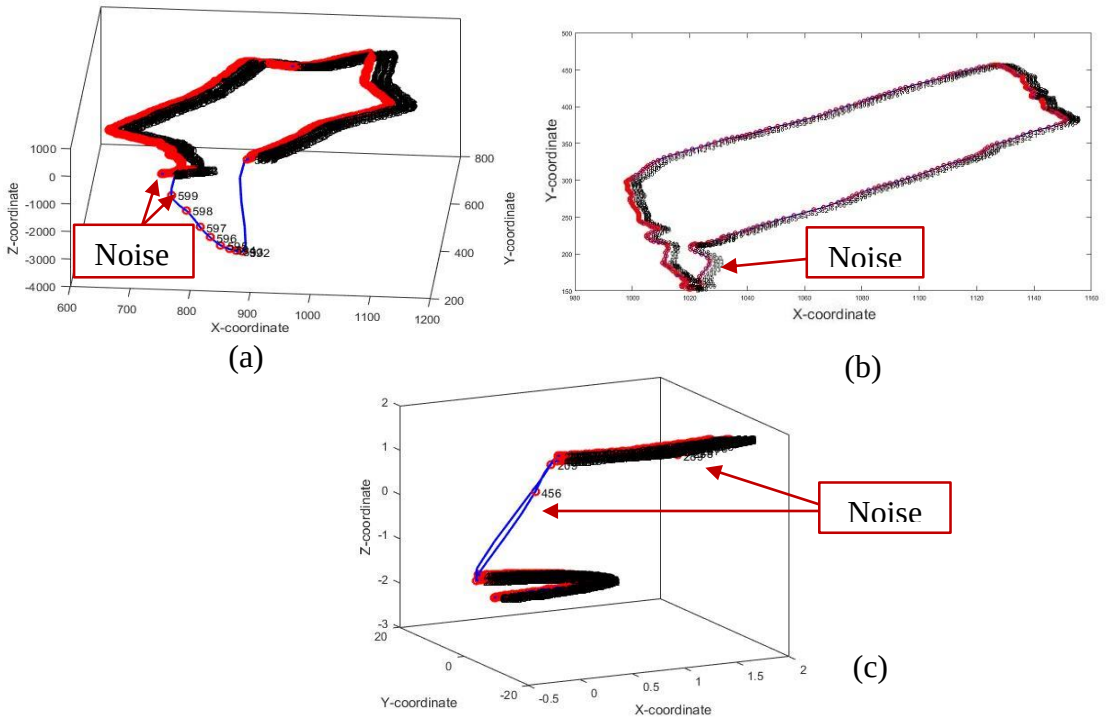
The present study is conducted to measure the feasibility of the B-code generation program through some target shapes. Figure 51 demonstrates the general procedure adopted in this study, with three different Matlab programs have been explored for the data filtering, 3D point selection and also B-code generation, respectively. The first phase of this data processing includes data acquisition, data filtering and data selection. Therefore, each of this process would be discussed accordingly. In this study, three random shapes

have been selected, as depicted in Figure 52. Data acquisition is conducted by using a magnetic sensor, as previously described in Section 5.1. This type of data digitizing induces unwanted noise, as illustrated in Figure 53. Therefore, the following section demonstrates how the introduced programs of the 3D point cloud data filtering and 3D data point selection are used to process this type of input data for the subsequent 3D point selection program.



(a) (b) (c)

Figure 52: Models of this study; (a) star box, (b) phone box and (c) dental cast



(a) (b) (c)

Figure 53: Initial data for each model (a) star box, (b) phone box and, (c) dental cast

The sensitivity of the magnetic sensor has resulted in a collection of noise for each model, as illustrated in Figure 53. A brief example of the process to filter the data has

been elaborated in Figure 54. For example, in Figure 54a, the required arch contour is indicated in the circle while the remaining data is considered as noise. During the first stage of this data filtering procedure, the outliers are considered as noise and must be discarded. A parametric fitting of points is the method used to identify the outliers. In this method, the points are fitted through parametric fitting cubic spline curve. Through the interpolation, the noisy data could be easily recognized and deleted. This is possible since the interpolation of point will indicate each point with its index number according to the input file, as illustrated in Figure 54c. Therefore, the unwanted point number could be identified and erased in the original input document, providing a data with no noise for the subsequent point selection stage. The detection of unwanted noise is feasible with the ability to zoom, rotate and move the figure during the data filtering procedure. On the other hand, a double dental arch data is recorded by the digitizer as shown in Figure 54b, in addition to the noise and therefore the user intervention in manually choosing the desired arch curve is executed, creating the required noise free data as depicted in Figure 54c.

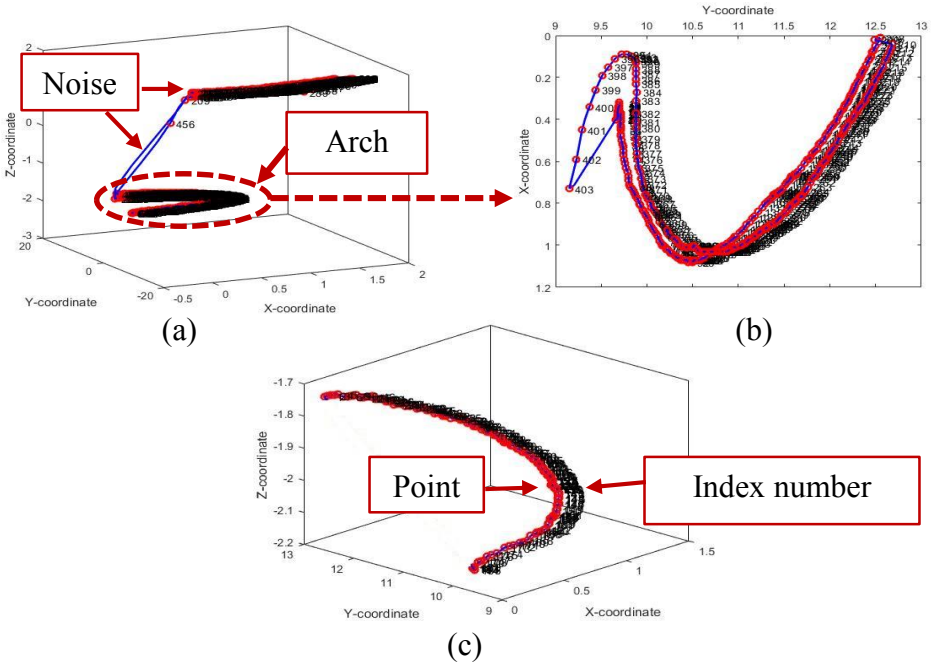
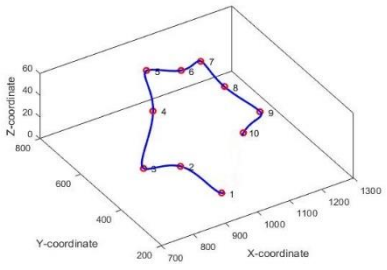
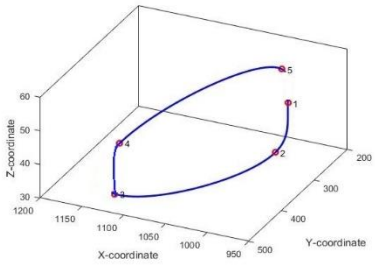


Figure 54: Graphical illustration of data filtering procedure; (a) the initial data of the model which clearly shows the noisy data above the desired feature (in +z direction), (b) is the 2D view of the target feature after the first stage of data filtering and, (c) is the noise free data

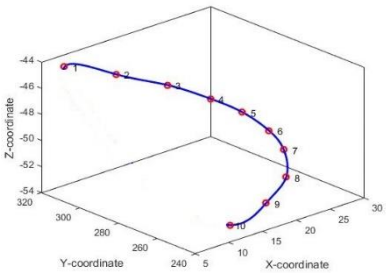
The filtered point cloud data from the previous 3D filtering procedure would be further processed by the second program, which is the 3D point selection. A brief introduction about the program and its interface has been elaborated in Section 5.2.2. The program allows the user to select the desired point cloud data by just clicking on the right point in 3D interface. The dataset could also be rotate, zoom and move to make the selection process more feasible. In this method, the selection of point is defined manually by the user and not automatically computed by the program. Therefore, the user must decide which points to be considered for the B-code generation. In the star box model, for example, the selection of point is straightforward where the intersection of lines in each edge is supposedly chosen as the bend point. The dental cast, however; consists of a curve and as compared to the lingual arch-wire case in dentistry, the bend point must be determined manually. In this present work, the selection of bend point for this model is made arbitrarily with a consideration to theoretically demonstrate the proposed procedure, rather than to practically apply in the lingual arch wire treatment planning. In this context, the expertise of the dental technician in making the decision is required due to the maximum accuracy of the treatment. Figure 55 exhibits the selected 3D points for each model, in addition to the XYZ table for each. The XYZ tables are generated automatically as the output of this program, which are subsequently adopted for the B-code generation program to establish the bending parameters (L, β, θ).



Point	x	y	z
1	897.226	215.792	25.4762
2	839.06	325.012	38.1762
3	753.462	372.764	37.4142
4	876.144	519.068	53.4162
5	952.344	671.722	57.4802
6	1055.214	665.626	47.8282
7	1157.576	728.872	35.3822
8	1157.576	611.778	31.0642
9	1207.36	513.988	18.8722
10	1095.092	419.5	27.0002



Point	x	y	z
1	1005.176	240.792	44.704
2	997.81	297.434	35.56
3	1127.604	456.438	30.988
4	1150.718	381.762	38.1
5	1022.956	213.106	51.308



Point	x	y	z
1	7.112	315.468	-44.45
2	12.446	307.34	-45.466
3	17.78	299.466	-46.736
4	21.844	291.338	-48.006
5	24.384	283.718	-49.022
6	26.162	276.098	-50.292
7	26.162	268.478	-51.308
8	24.13	260.096	-52.578
9	19.304	253.492	-53.34
10	13.208	250.952	-53.848

Figure 55: Completed pre-selection procedure for each model, in addition to the generated XYZ table. The star box consists of (npts=10 points), the phone box has (npts=5points), while the dental cast has (npts=10 points)

7.2 B-code generation process from 3D point cloud data

The final aim of the study is to bend a wire conforming to the B-code data retrieved from a computer. In order to automate the bending operation with a CNC machine, the CAM data has to be prepared (L, β, θ). The B-code table as summarized in Table 10 has been generated for all models by the developed program through some mathematical formulae as briefly explained Section 5.4.

Table 10: B-code table for each test model

Model	*No. of bend (nth)*	*L (mm)*	*β (degrees)*	*θ (degrees)*
Star box (npts=10)	1	124.39	0	33.35
	2	98.02	173.40	100.82
	3	191.60	-165.66	13.86
	4	170.66	159	67.12
	5	103.50	-173.59	34.93
	6	120.97	174.81	121.24
	7	117.17	-178.43	27.23
	8	110.41		77.36
		146.96		
Phone box (npts=5)	1	57.85	0	47.05
	2	205.30	-5.87	123.55
	3	78.49	2.58	54.19
		212.00		
Dental cast (npts=10)	1	9.77	0	1.84
	2	9.60	113.81	7.49
	3	9.18	6.91	8.09
	4	8.10	-27.15	5.62
	5	7.93	26.7	13.09
	6	7.69	-12.31	13.52
	7	8.72	8.66	22.57
	8	8.21		31.12
		6.62		

7.3 3D simulation of selected models

In order to measure the accuracy of the B-code (L, β, θ) in performing the desired wire bending operation for each model, 3D simulations are carried out. 3D simulation is chosen since the physical CNC bending machine has not been established yet. The input data for this simulation consists of the generated B-code. Theoretically, the feeding length (L) feeds the wire through a feeding roll towards the bending mechanism. The wire segment (L) is advanced by a sending mechanism to a wire bending station. As the forward

end of the severed segment enters the bending station, a stepping motor receives signals from a control system, which includes the microprocessor and computer, for rotating the turning mechanism (β) in either clockwise our counter clockwise and incrementally displacing a bend on the desired bend point in a controlled manner to produce a wire segment having the desired bending angle (θ). Figure 56 to Figure 58 illustrate the simulation result for each model.

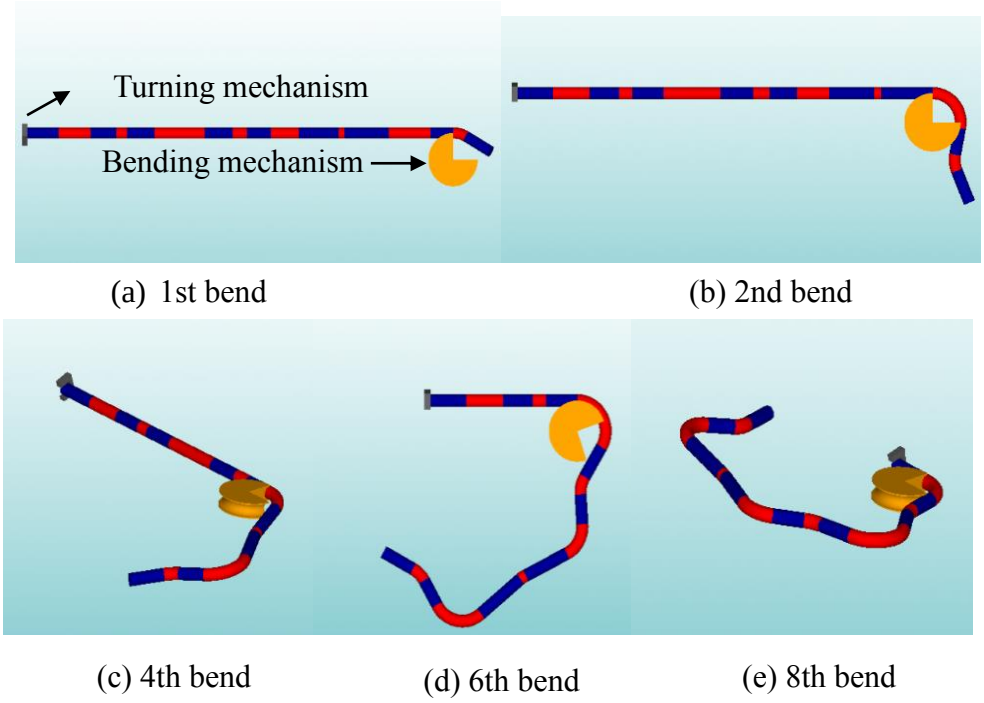


Figure 56: Wire bending simulation for the star box model

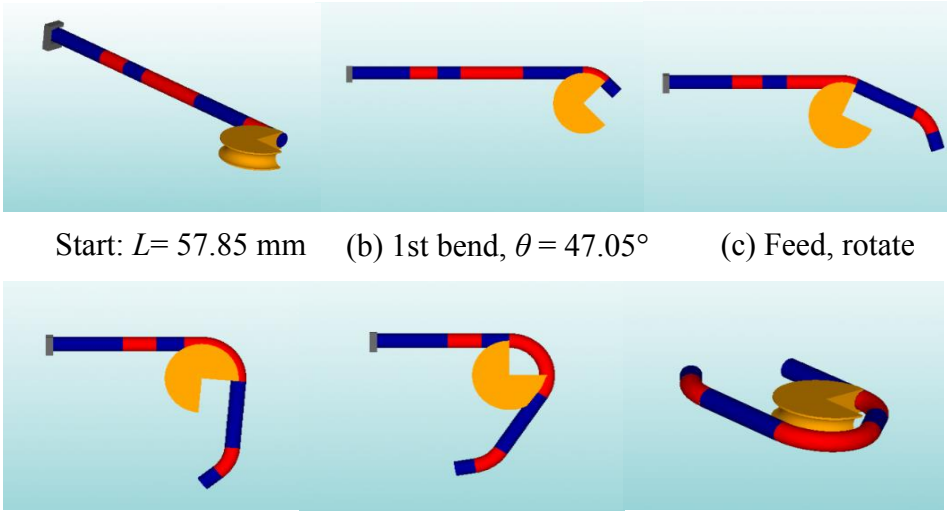


Figure 57: Wire bending simulation for the phone box model

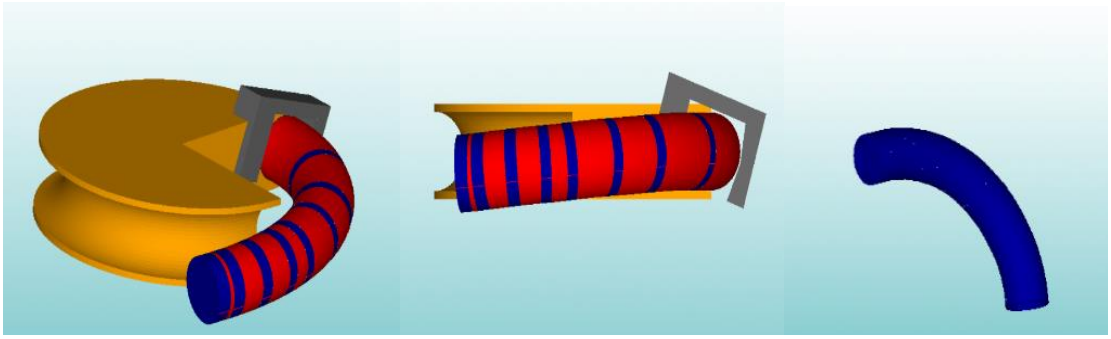


Figure 58: Wire bending simulation for the dental cast model

Throughout the observation, the calculated B-code (L, β, θ) was proven to be able to control the movement of sending mechanism, turning mechanism and bending mechanism, as theoretically expected. The initial straight wire was turned into the required shape in the end for each model, as briefly depicted in Figure 56-58. In the illustration, the wire sending mechanism is not clearly indicated, in contrast to the other two mechanisms, which are the turning mechanism and the bending mechanism. The radius of the roller, r is not considered for this simulation since the inputs given were only (L, β, θ). In the theory of 3D linear segmentation, the straight wire is bent into the desired shape through multiple 3D linear segments, which refers to the sharp bend instead (r = 0). However, in this simulation, a curve bent was observed. This might contradict to what the theory say but as mentioned earlier, the simulation was conducted just to validate the operational sequence of the desired parameters (L, β, θ). In this context, the radius (r) of the bending mechanism is insignificant. Therefore, the roller represents the bending mechanism and was used to check whether the given feed length (L) could control the movement of this roller to the exact bend location and perform the desired bending operation in accordance with (θ). On the other hand, the bend points are highlighted in red, with n^{th} = 8, n^{th} = 3 and n^{th} = 8 for the star box model, the phone box model and the dental cast model, respectively.

7.4 Algorithms for 2D image point digitization

In this procedure, a 2D point selection interface, as briefly introduced in Section 5.2.3 is used to plan the required bend points. In this section, the feasibility of the introduced program would be validated, in addition to the generated B-code data. For this reason, a random human dental arch image is used as a target shape. The purpose of this

recent work is to acquire the human dental arch curve through the introduced program, in reference to the human dental arch digitization procedure, as described by [73]-[74]. In this section, the procedure of using this interface for the 2D point selection would be graphically demonstrated, by using human lingual arch curve as an example.

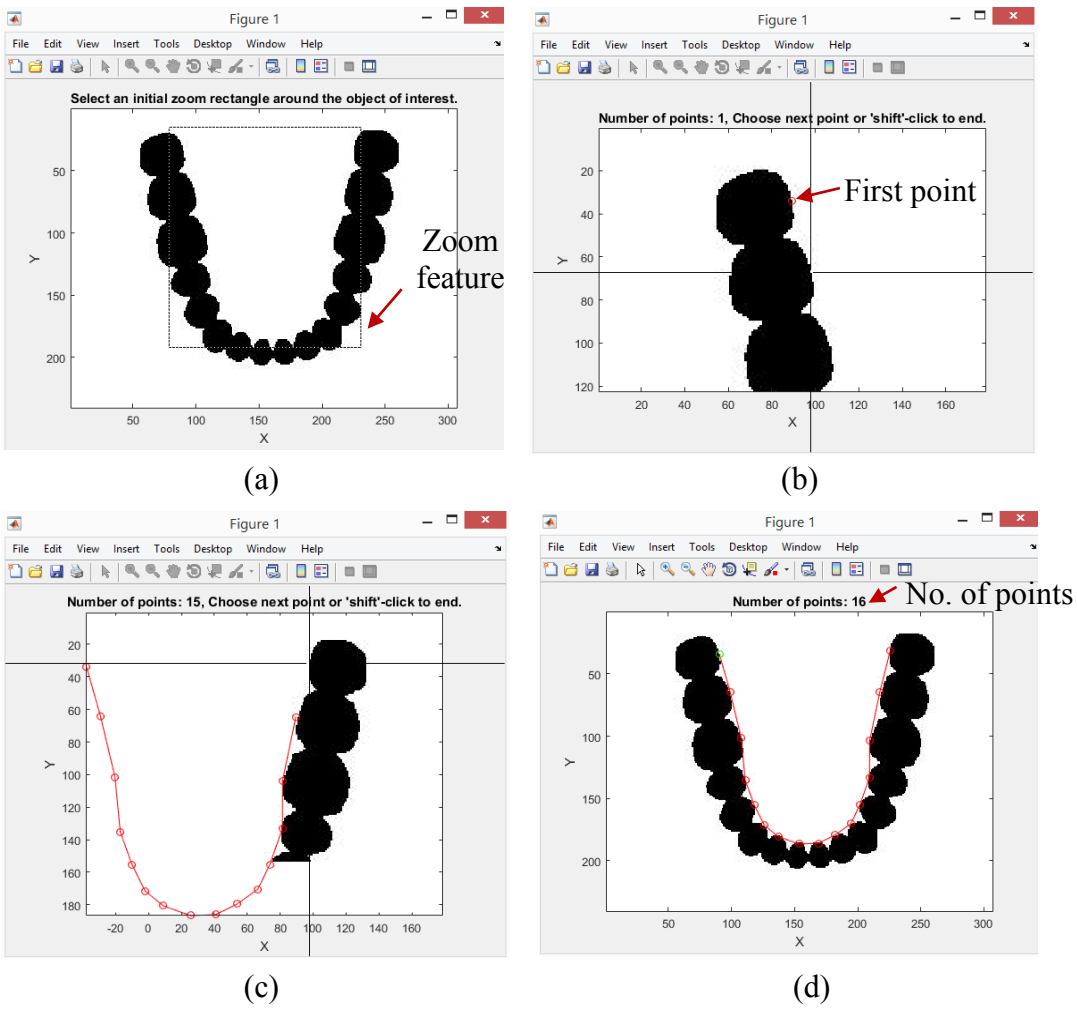


Figure 59: Procedure of using the 2D point selection interface

Firstly, the 2D image is uploaded to the interface, as illustrated by Figure 59(a). This interface allows the user to select an initial zoom around the object of interest. For example, in Figure 59(b), the object has been zoomed from the previous step. The first 2D point selection is indicated by the red circle. Then, the selection of point continues as shown in Figure 59(c). Once the desired point selection is accomplished, the user has to enter a 'shift', follows by clicking the last point. This generates the 2D interpolation of the selected points, as depicted in Figure 59(d). Also, the number of selected points would be printed on the screen. In this example, 16 points have been chosen. This example

demonstrates how the 2D point selection towards the 2D image data works. Several points have to be digitized to represent the anatomical of dental arch, as depicted in Figure 60. In this study, 18 points were digitized for the labial arch curve, which were the midincisal point of the incisors, the canine cusp tip, the buccal cusp tip of the premolars, and the mesiobuccal and distobuccal cusp tips of the first and second molars. In addition, 16 points were digitized for the lingual arch curve, which were the midincisal point of the incisors, the canines, the premolars and the first, second and third molars.

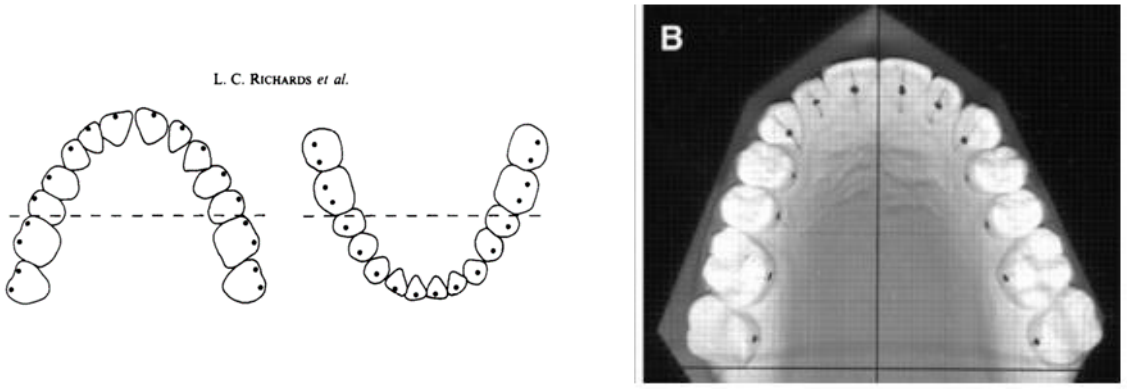


Figure 60: Reference for labial (left) and lingual [74] (right) point digitization

A review of different approaches used by other researchers for this human dental arch digitization has confirmed that the introduced 2D point digitization interface in this study is more straightforward. Richardson [73] used a Hewlett-Packard 9874A digitizer, then transferred the digitized image into a main-frame computer where the coordinate set was first to bring the contact point between the translated incisors to the origin, then rotated to align the most mesial points of the first molars with the x-axis. In addition, Noorozi [75] has shown a cumbersome procedure in generating the XY point coordinates, where the digital image of occlusal casts were prepared by scanning and subsequently transferred to AutoCAD in order to get the desired coordinates. In contrary to this present work, the procedures are slightly different where the desired 2D image data of each sample is uploaded to the program and the user could directly select the desired 2D points in a real time environment and after the selection process is completed, the XY coordinates of those pre-selected points will be automatically generated.

In this study, another program to process the 2D image input data, namely the image segmentation program is additionally explored. The reason for experimenting with this program is to initially extract the human dental arch figure from the 2D image, before

the 2D point selection is used. This segmentation procedure helps to prepare the image in order to get a more accurate human dental arch curve. The adoption of a localizing region-based active contour by [76] is used for this purpose. Details about the program could be further referred in the website. In order to run this program, the following information are provided at first and the description of each script has been briefly indicated.

```
I = imread('teeth.jpg');  %-- load the image
m = zeros(size(I,1),size(I,2));  %-- create initial mask
m(111:222,123:234) = 1;
I = imresize(I,.5);  %-- make image smaller
m = imresize(m,.5);  %     for fast computation
subplot(2,2,1); imshow(I); title('Input Image');
subplot(2,2,2); imshow(m); title('Initialization');
subplot(2,2,3); title('Segmentation');
seg = region_seg(I, m, 6350); %-- Run segmentation
subplot(2,2,4);     imshow(seg);     title('Global     Region-Based
Segmentation');
```

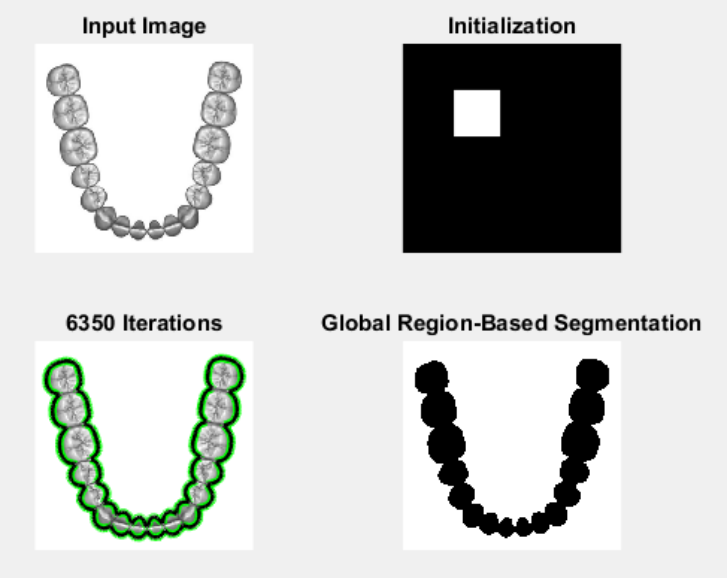


Figure 61: Localizing region-based active contour for extracting dental arch feature

Once the program has been run, the image segmentation process takes place accordingly. In this procedure, the accuracy of contour extraction depends on the number of iteration that is defined earlier. Figure 61 shows the outcome in using 6350 times of iterations, resulted to a perfect human dental arch feature extraction for both labial and

lingual arch curve. This output is subsequently used for the 2D point selection program which will be subsequently explained.

7.4.1 Human dental arch curve point digitization procedure

This section illustrates the graphical sequence in planning the required bend points with regard to Figure 60 by using the labial arch curve extraction as an example. The 2D point digitization interface as indicated by Figure 62 allows the user to click the desired point on the uploaded image. In this labial arch curve, for example, 18 points for the subject of maxillary normal occlusion teeth have been selected. Table 11 depicts the XY table for those selected points, which is then employed for the B-code generation program. Table 12 summarizes the B-code results for this curve, in relation to (L, β, θ).

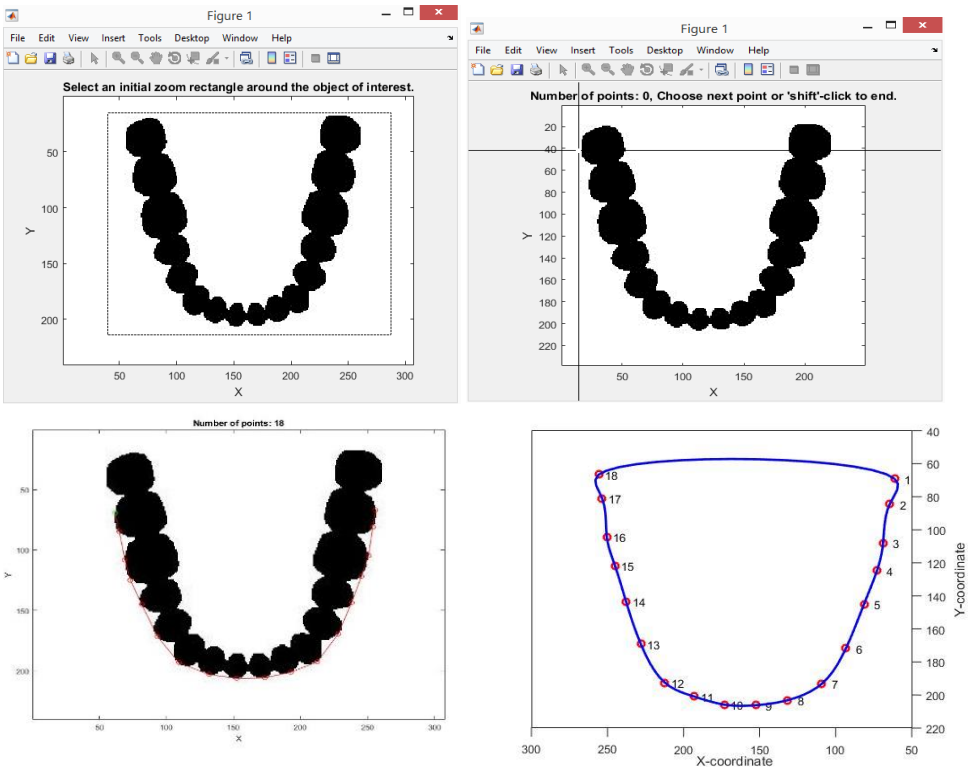


Figure 62: Point digitization example for the labial arch curve

The similar procedure as demonstrated in the previous section is executed for the generation of lingual arch curve. In this labial arch curve, 16 points (with the addition of third molars) for the subject of maxillary normal occlusion teeth have been selected. Table 13 shows the extracted XY Cartesian coordinates of those 16 points while Table 14 summarizes the B-code results for this curve, in relation to (L, β, θ). Therefore, the

generated B-code table for both labial and lingual arch curve would be used in the 3D simulation in the following section.

Table 11: XY table (labial)

Point	*X*	*Y*
1	61.6465	69.11059
2	64.86011	84.59546
3	69.12193	108.1607
4	73.0983	124.8327
5	81.53308	144.9991
6	93.31096	171.672
7	109.1824	193.2533
8	131.794	203.2713
9	152.5732	206.3193
10	173.2492	205.9953
11	192.9252	201.1355
12	212.4144	192.7118
13	228.1734	169.4565
14	238.0109	143.6115
15	245.2859	122.0624
16	250.7339	104.621
17	253.854	81.2344
18	256.0625	66.83685

Table 12: B-code table (labial)

No.	*L*	*β*	*θ*
Bend 1	15.81482	0	1.473044
Bend 2	23.94752	0	3.163513
Bend 3	17.13964	0	9.282802
Bend 4	21.8593	0	1.127122
Bend 5	29.15754	0	12.50703
Bend 6	26.78909	0	29.7727
Bend 7	24.73145	0	15.55066
Bend 8	21.00156	0	9.242699
Bend 9	20.67854	0	12.97611
Bend 10	20.26728	0	9.501278
Bend 11	21.23176	0	32.50123
Bend 12	28.0919	0	13.28505
Bend 13	27.65394	0	2.183815
Bend 14	22.744	0	1.30801
Bend 15	18.27247	0	9.747545
Bend 16	23.59381	0	1.121677
	14.56595		

Table 13: XY table (lingual)

Point	*X*	*Y*
1	90.37172	33.92274
2	98.90783	64.37097
3	107.6443	101.5335
4	110.8732	135.2318
5	118.0009	155.3255
6	126.0283	171.5405
7	137.0397	180.5745
8	153.4601	186.5972
9	168.8578	185.9949
10	182.051	179.37
11	194.4868	170.336
12	201.6914	155.2794
13	209.4072	132.9955
14	209.4749	103.649
15	217.5292	64.57872
16	225.835	31.15449

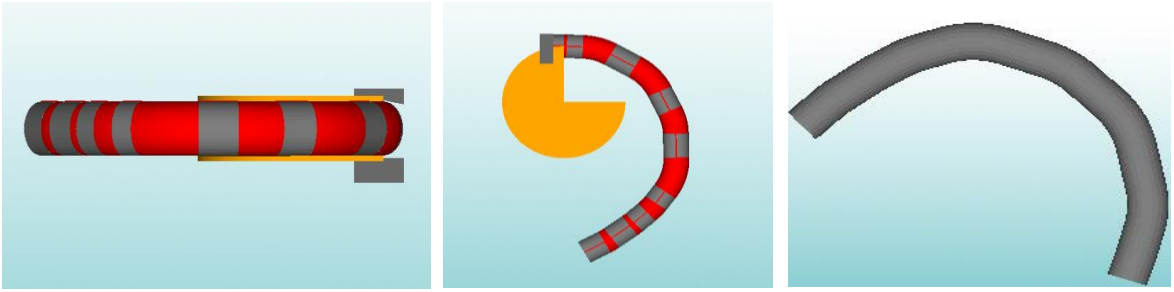
Table 14: B-code table (lingual)

No.	*L*	β	θ
Bend 1	31.62214	0	2.431417
Bend 2	38.17567	0	7.756145
Bend 3	33.85259	0	14.05754
Bend 4	21.3204	0	6.807534
Bend 5	18.09334	0	24.29536
Bend 6	14.24298	0	19.22438
Bend 7	17.49005	0	22.38191
Bend 8	15.40951	0	24.42346
Bend 9	14.76314	0	9.333033
Bend 10	15.37087	0	28.43271
Bend 11	16.69156	0	6.472282
Bend 12	23.58186	0	18.96659
Bend 13	29.3466	0	11.51629
Bend 14	39.89182	0	2.306924
	34.44076		

7.4.2 3D simulation of dental arch curves

In this section, the 3D simulation of the digitized points with regard to the labial arch curve and the human arch curve would be demonstrated. The generated XY tables from the 2D point selection program, as shown in Table 11 and Table 13 are used to feed the B-code generation program. Then, the generated B-code table as indicated in Table 12 and Table 14 are used for the 3D simulation, similar to the 3D simulation in Section 7.3.

Figure 63 shows the result of this B-code simulation, which clearly demonstrates that this is a 2D wire bending through the use of working planes. The B-code controls the simulation by feeding the L value, which is subsequently bend by the bending roller in accordance with the value of θ. The process continues until the required 2D wire bending is completed. The similar procedure is repeated for the lingual arch curve fabrication, as shown in Figure 64.



(a) (b) (c)

Figure 63: 3D simulation of labial arch curve (a) side view (b) top view (c) finished part

Throughout the observation, the calculated B-code (L, β, θ) is proven to be able to control the movement of sending mechanism and the bending mechanism, as theoretically expected. The initial straight wire was turned into the required shape in the end for each model, as briefly depicted in Figure 63-64. In the illustration, the wire sending mechanism is not clearly indicated, in contrast to the other two mechanisms, which are the turning mechanism and the bending mechanism. The radius of the roller, r is not considered for this simulation since the inputs given were only (L, β, θ). In the theory of 3D linear segmentation, the straight wire is bent into the desired shape through multiple 3D line segments, which refers to the sharp bend instead (r = 0). However, in this simulation, a curve bent was observed. This might contradict to what the theory say but as mentioned earlier, the simulation was conducted just to validate the operational sequence of the

desired parameters (L, β, θ). In this context, the radius (r) of the bending mechanism is insignificant. Therefore, the roller represents the bending mechanism and was used to check whether the given feed length (L) could control the movement of this roller to the exact bend location and perform the desired bending operation in accordance with (θ).

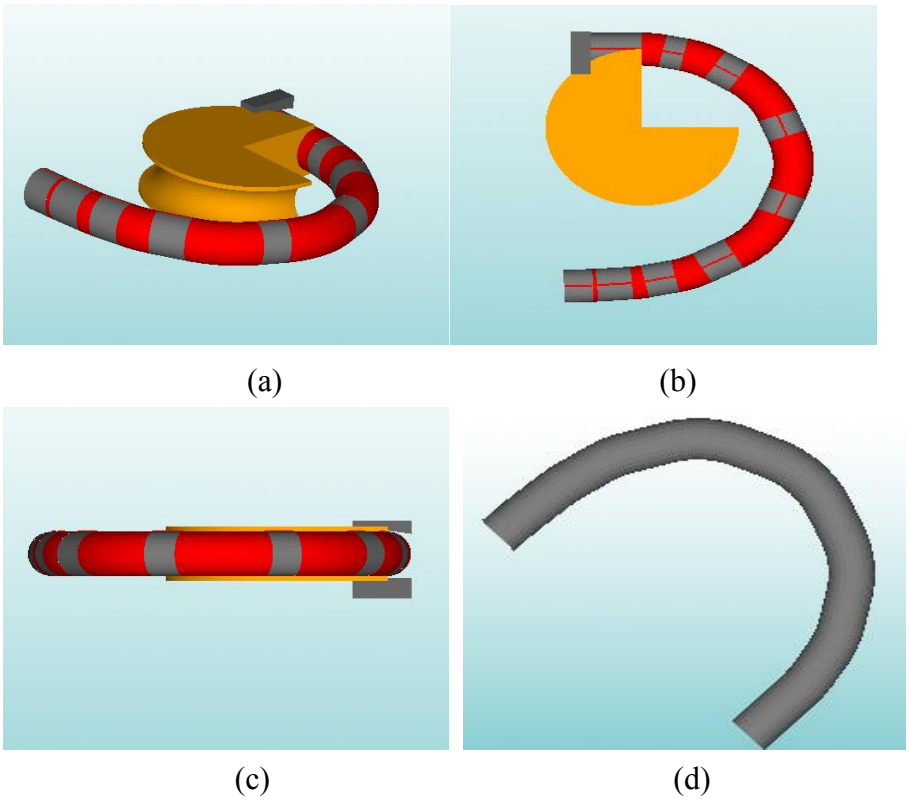


Figure 64: 3D simulation of lingual arch curve (a) isometric view (b) top view (c) side view (d) finished part

7.5 Summary of 3D simulation results

To conclude, a different approach in generating the digital wire design in preparing the CAM data for the subsequent B-code generation has been validated in this present work through the use of 3D point cloud data and the 2D image data as inputs. In dealing with 2D image as the input data, two procedures have been proposed, which is to initially execute the 2D image segmentation and follows by the 2D point selection method for the purpose of getting a more accurate arch curve. In addition, the demonstration of using the introduced programs, which are a 3D filtering and 3D point selection have also been successfully demonstrated through some examples. The B-code tables for all cases have been validated through 3D simulations.

CHAPTER 8

SEMI-OPTIMIZED ARCH CURVE DESIGN FOR B-CODE GENERATION

In this chapter, a mathematical definition of human dental arch form has been additionally investigated, in an attempt to design an optimized arch shape for the braces treatment. This could improve the theory of the 3D linear segmentation into a more semi-optimized curve. However, the actual arch-wire bending operation to realize the semi-optimized arch shape has not been studied, which remains as the recommendation for future work.

8.1 Mathematical definition of dental arch form

Dental arch is an important element in orthodontics and is a fundamental principle in orthodontic planning and therapy [77]-[78]. Dental arch shapes for humans have been studied and attempts to provide mathematical description of dental arches have been done by several researchers since the first quarter of twentieth century through several mathematical functions like polynomial, cubic, catenary curve and beta function. Many methods have been developed to describe dental arch morphology and of the previously described methods, the application of polynomial has generated interest in many researchers, focusing on the fourth-order polynomials and the sixth-order polynomials. Triviño [79] has proposed the use of sixth-degree polynomial equation as the function that best described dental arch configuration which is supported by Noroozi [75]. On the other hand, Lu [80] has started the definition of dental arch by using the fourth-order polynomial and has been further supported by Alharbi [81] and Richard [73]. To relate this mathematical definition with the orthodontics treatment, Muhammad [82] had suggested that the fourth-order polynomials could be effectively used to define a smooth dental arch curve which could further be a fixed applied into fabricating custom arch wires or orthodontic apparatus, which could substantially aid the treatment planning during the bend points planning process. In addition, due to its advantage in providing a more naturally smooth curve, the fourth-order polynomials function maybe used to fabricate customized arch-wires, or even an entire fixed orthodontic system [81]. Therefore, in this

chapter, the previously generated XY-tables for both the lingual and the labial arch curve from Section 7.4.1 would be subsequently employed to design a semi-optimized curve through fourth-order polynomial and sixth-order polynomial. In addition, the mathematical definition of the curve would be disclosed with regard to both the fourth-order and the sixth-order polynomials in the curve fitting interface. The aims for this recent analysis are to explore an alternative way to generate a more semi-optimized curve for the case of arch-wire bending and to identify which order of polynomial is better in describing the human dental arch form. Results of this mathematical representation would be briefly discussed in the following section, detailing on the R-square and the adjusted R-square values of the fitting. The subject of this study is a maxillary with a normal occlusion.

8.2 Polynomial fitting of the XY-table

Curve fitting in Matlab has been used to fit the 2D points to the polynomial function of fourth order and sixth order by least square error method. The fourth-degree polynomial and the sixth-order polynomial were selected to establish the dental arch form, in reference to the literature. The Cartesian coordinates (X, Y) of the sample were plotted in Matlab to obtain the fourth-degree polynomial and the sixth-order polynomial function for each curve segment and the graphic representation. For this analysis, the XY-tables of the labial arch curve and lingual arch curve, which are discussed in Section 7.4.1 are used to feed the program. The program fits standard polynomial equations (degrees four and eight inclusive) to the data points. These polynomial equations are in the form of:

$$Y = \sum_{i=0}^{8} a_i \, x^i \tag{13}$$

The selected curve was fit by least square error methods, which the program was instructed to approximate the coefficients of the polynomial equation of each degree such that:

$$\sum_{i=1}^{n} (y_i - \hat{y}_i) = 0 \tag{14}$$

In Equation 14, n is the number of data points, $\mathbf{y_i}$ refers to the Y-coordinate of the data point (x_i, y_i) on the common line of occlusion and $\hat{\boldsymbol{y}}_{\boldsymbol{i}}$ is the computed Y-coordinate of the point on the polynomial curve corresponding to the original x_i. The mean square errors are as shown in Equation 15.

$$\sum_{i=1}^{n}(\boldsymbol{y_i} - \hat{\boldsymbol{y}}_{\boldsymbol{i}})^{\mathbf{2}} = \mathbf{0} \tag{15}$$

A series of data fitting were conducted using this program, by adopting the XY-table data belongs to the lingual and the labial arch curve from Section 7.4.1. This is explored in order to determine how accurately the fourth-degree and the sixth-degree curve fit the data points. Results of this data fitting would be discussed, in brief with each function is represented by a mathematical equation in consideration to this data fitting. For the labial arch curve, the sixth-order polynomials describe the curve better than the fourth-order polynomials with R-square of 0.9979 and the adjusted R-square of 0.9968, as illustrated in Figure 65.

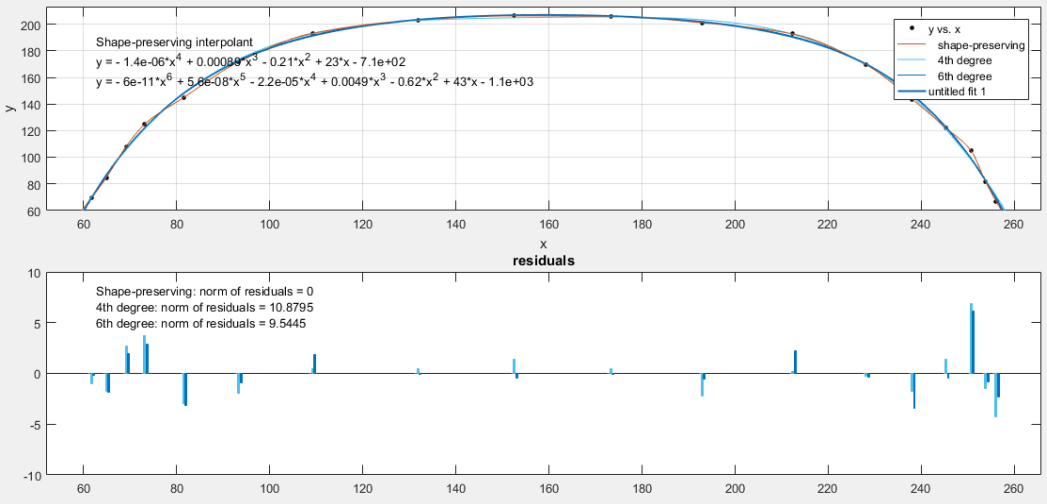


Figure 65: Residuals plot for the comparison of fourth degree polynomials and the sixth degree polynomial with shape reserving curve of the generated labial arch curve

For the lingual arch curve, the sixth-order polynomial describes the curve better than the fourth-order polynomial, with the R-square value of 0.9836 and the adjusted R-

square of 0.9726. For this brief analysis, the sixth-order polynomial could fit the subject, which is a maxillary with a normal occlusion. This indicates that to design the treatment planning for the orthodontic treatment, perhaps the sixth-order polynomial could be considered as a reference line during the planning of the bending point's process. Therefore, and a new approach to generate the B-code in accordance with this mathematical function has to be further investigated, by using more human dental arch curves to examine the best mathematical definition which perfectly describes the arch form, with regard to specific community or country. This allows a better treatment planning for the orthodontic treatment by suggesting the optimized arch curve for each patient which could be used as a reference during the bend points planning stage.

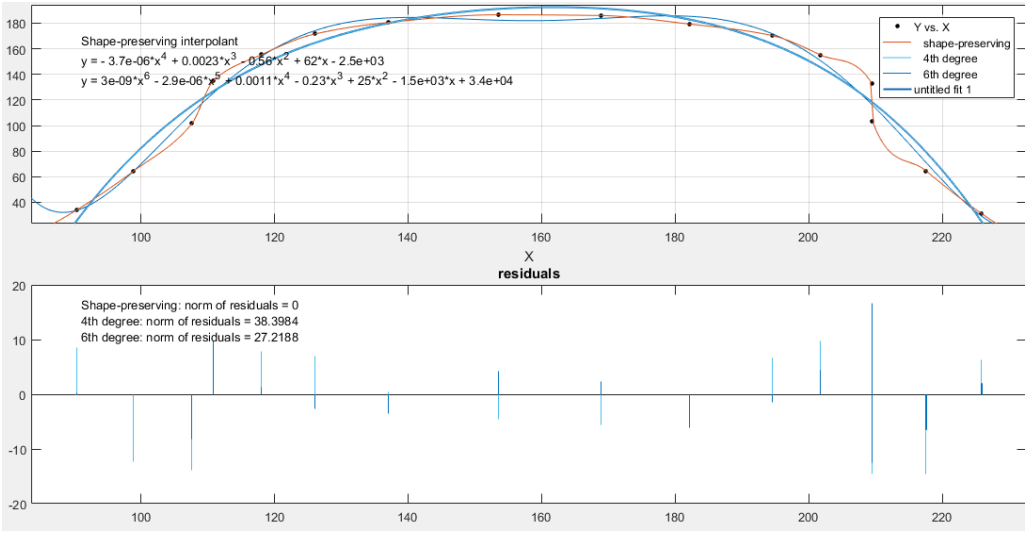


Figure 66: Residuals plot for the comparison of fourth degree polynomials and the sixth degree polynomial with shape reserving curve of the generated lingual arch curve

CHAPTER 9

CONCLUSIONS & FUTURE WORK

This research proposes a concept for the automation of dentistry wire bending operation through a CNC wire bending machine. A review of dentistry wire bending systems for the past 3 decades have indicated some gaps in terms of the application of the automated wire bending systems which focus on the arch-wire manufacturing, in addition to the less exploration in using the CNC wire bending machine technology in comparison to the robotic device. Therefore, taking some information related to challenges among dental experts who resist to accept the new technology compared with fresh graduates due to the complexity of the technology and the high cost of the device, therefore a system which is straightforward, less-expensive and compatible with other devices has to be considered. Therefore, a CNC wire bending machine is chosen to realize the wire bending operation, and some additional factors have to be incorporated into the machine, such as the portability and it has to be a desktop size, which is suitable to be used during the chair-side treatment.

In order to do so, a theory of a 3D linear segmentation has been introduced to represent the freeform curve into a multiple 3D linear segments. A freeform curve is hard to be defined and the linear segmentation allows the bending operation for the desired shape to be performed through the bend of multiple 3D linear segments. Relative spatial parameters between adjacent 3D linear segments have been analyzed which led to the introduction of some mathematical formulae to theoretically calculate the parameters. This relates into the significant wire bending parameters to realize the actual 3D wire bending operation. In order to verify the important wire bending parameters, a shape definition analysis through a sheet metal feature has been carried out in the beginning of study. From this analysis, the most required wire bending parameters have been determined, and the proposed mathematical formulae have been validated through a comparison of wire bending parameters between the sheet metal and the proposed theory.

The formulae was further used for the development of the B-code generation program, which supports the idea that the movement of the machine is composed from a

sequential movement of its mechanisms, in terms of the feeding mechanism, the rotating mechanism and also the bending mechanism. Each mechanism is controlled by the calculated B-code data, referred as (L, β, θ). In order to prepare the data for the B-code generation program with regard to different cases, a few approaches have been explored to process various type of input data, in terms of the teeth scan (STL file), the 3D point cloud data and also the 2D teeth image. Each input has to be processed in a slightly different way for the extraction of the XYZ Cartesian coordinates of 3D points and adopts the similar interface to generate the B-code. For that reason, several interfaces have been introduced for the desired procedure, such as an automatic vertices coordinate extraction from the IGES file, a data filtering procedure to identify the outliers in 3D point cloud data, a 3D point selection program and also a 2D polyline selection from 2D teeth image.

To conclude, the research objectives in this study have been achieved. Firstly, a concept of automating the dentistry wire bending operation has been proposed by adopting CNC wire bending machine to realize the wire bending operation through a concept of feed, rotate and bend. Secondly, the B-code generation method has been successfully developed in Excel and Matlab, which theoretically demonstrates how the design information data could be translated into the machine language through some mathematical formulae. Finally, the feasibility of the introduced B-code generation program has been validated and demonstrated through some examples. Therefore, the ultimate findings have been achieved, focusing on the introduction of bending points' algorithm and the theoretical manufacturing of personalized target shape. As for future work, the interface to directly integrate the B-code generation program to the CNC wire bender has to be developed, in addition to the planning of wire spring back algorithm. In order to do so, the material properties of each dental wire from different manufacturers have to be collected and simulated in the FEA software. In addition to that, a database which could store all of the wire design templates could also be established in the future, in line with the recent introduction of the internet of things (IoT).

REFERENCES

[1] A. Gilbert, "Robots for the elaboration of lingual archwires," *United States Patent*, 2011.

[2] G. Davidowitz and P. G. Kotick, "The use of CAD / CAM in dentistry," *Dent. Clin. North Am.*, vol. 55, no. 3, pp. 559–570, 2011.

[3] D. Rekow, "Computer-aided design and manufacturing in dentistry: A review of the state of the art," *J. Prosthet. Dent.*, vol. 58, no. 4, pp. 512–516, 1987.

[4] J. C. Davenport, R. M. Basker, J. R. Heath, J. P. Ralph, P. O. Glantz, and P. Hammond, "Clasp design," *Br. Dent. J.*, vol. 190, no. 2, pp. 71–81, 2001.

[5] R. A. Hamid and T. Ito, "A proposal of wire bending mechanism for dentistry applications," in *International Design and Concurrent Engineering Conference 2015 (iDECON 2015)*, Tokushima, Japan, 2015, Paper No.10.

[6] J. Jiang, Y. Zhang, C. Wei, T. He, and Y. Liu, "A review on robot in prosthodontics and orthodontics," *Adv. Mech. Eng.*, vol. 7, no. 1, pp. 1–21, 2014.

[7] R. C. L. Sachdeva, S. L. T. Aranha, M. E. Egan, H. T. Gross, N. S. Sachdeva, G. F. Currier, and O. Kadioglu, "Treatment time: Suresmile vs conventional," *Orthod.*, vol. 13, pp. 72–85, 2012.

[8] T. Ito, R. A. Hamid, and T. Ichikawa, "Collaborative design and manufacturing of prosthodontics wireclasp," in *23rd ISPE Inc. International Conference on Transdisciplinary Engineering*, 2016, vol. 4, pp. 421–428.

[9] Zhang, "Research of 3D virtual design and automated bending of oral orthodontic archwire," *Int. J. Adv. Comput. Technol.*, vol. 5, no. 8, pp. 741–749, 2013.

[10] T. Ito, R. A. Hamid, M. W. Abd Rashid, and E. Osman, "TMAC Design Workshop Training Seminar 2016," in *JSME 26th Design and Systems Conference*, Tokyo, Japan, 2016, Session ID: 1401. [online]: http://doi.org/10.1299/jsmedsd.2016.26.1401.

[11] R. A. Hamid, K. Morisaki, and T. Ito, "B-code generation for a CNC dentistry wire bending mechanism," in *Proceedings of International Symposium on Scheduling (ISS 2017)*, 2017, pp. 105–110.

[12] R. A. Hamid and T. Ito, "Automatic extraction of vertices coordinates for CNC code generation for dental wire bending," *Int. J. Agil. Syst. Manag.*, vol. 10, no. 3/4, pp. 321–339, 2017.

[13] R. A. Hamid and T. Ito, "A review of computer-aided wire bending systems for prosthodontics and orthodontics," in *5th JSME Chugoku_Shikoku Branch Conference*, Ehime, Japan, 2016, Session ID: 505 [online] http://doi.org/10.1299/jsmecs.2016.54._505-1_

[14] R. A. Hamid and T. Ito, "A review of recent advancements in automated dental wire bending," in *JSME Manufacturing Systems Division Conference 2018*, Tokyo, Japan, 2018, Paper No. 301.

[15] R. A. Hamid and T. Ito, "3D prosthodontics wire bending mechanism with a linear segmentation algorithm," *J. Adv. Manuf. Technol.*, vol. ISSN: 1985, Special Issue

(TMAC) Symposium 2016, pp. 33–46, 2016.

[16] R. A. Hamid and T. Ito, "Computer-aided wire design using sheet metal feature for dentistry applications," in *International Design and Concurrent Engineering Conference (iDECON 2016)*, Langkawi, Malaysia, 2016, Paper No. 104.

[17] R. A. Hamid and T. Ito, "Verification of sheet metal based wire bending procedures," in *JSME 26th Design and Systems Conference*, Tokyo, Japan, 2016, Session ID: 1403, [online] http://doi.org/10.1299/jsmedsd.2016.26.14.

[18] R. A. Hamid and T. Ito, "Shape definition and parameters validation through sheet metal feature for CNC dental wire bending," *Int. J. Comput. Aided Eng. Technol.*, (in-press).

[19] R. A. Hamid, K. Morisaki, and T. Ito, "Integration of CAD/CAM in developing CNC dental wire bending machine," *J. Adv. Mech. Des. Syst. Manuf.*, Special Issue: Advanced Production Scheduling (ISS2017:JAMDSM), (under-review).

[20] R. A. Hamid and T. Ito, "Computer-aided wire design using sheet metal feature for dentistry applications," *J. Adv. Manuf. Technol.*, vol. 11, No.1 (5th International Conference on Design and Concurrent Engineering (iDECON 2016), (in-press).

[21] R. A. Hamid and T. Ito, "Feasibility study on B-code generation method for CNC dental wire bending operation," in *International Design and Concurrent Engineering Conference 2017 (iDECON 2017)*, Osaka, Japan, 2017, Paper No. 7.

[22] R. A. Hamid and T. Ito, "Image segmentation to design semi-optimized curve for B-code generation," in *JSME 27th Design & Systems Conference*, 2017, Shimonoseki, Japan, Session ID: 2502.

[23] R. A. Hamid and T. Ito, "Feasibility study on B-code generation method for CNC dental wire bending operation," *J. Adv. Mech. Des. Syst. Manuf.*, Special Issue: Design and Concurrent Engineering Challenges for Sustainable Smart Manufacturing, (under-review).

[24] A. Gilbert, "An in-office wire-bending robot for lingual orthodontics.," *J. Clin. Orthod.*, vol. 45, no. 4, pp. 230–234, 2011.

[25] M. W. Brown, L. Koroluk, C. C. Ko, K. Zhang, M. Chen, and T. Nguyen, "Effectiveness and efficiency of a CAD/CAM orthodontic bracket system," *Am. J. Orthod. Dentofac. Orthop.*, vol. 148, no. 6, pp. 1067–1074, 2015.

[26] M. S. Bilgin, E. N. Baytaroğlu, A. Erdem, and E. Dilber, "A review of computer-aided design/computer-aided manufacture techniques for removable denture fabrication," *Eur. J. Dent.*, vol. 10, no. 2, pp. 286–291, 2016.

[27] R. Van Noort, "The future of dental devices is digital," *Dent. Mater.*, vol. 28, no. 1, pp. 3–12, 2012.

[28] I. Hutchinson and J. Y. Lee, "Fabrication of lingual orthodontic appliances: past, present and future.," *J. Orthod.*, vol. 40 Suppl 1, no. April, pp. S14-9, 2013.

[29] A. S. K. Salamah and M. S. Drogomyretska, "A glance about the applications of robot in orthodontics," *Int. J. Innov. Sci. Res.*, vol. 22, no. 1, pp. 178–182, 2016.

[30] P. Kumar, "Future advances in robotic dentistry," *J. Dent. Heal. Oral Disord. Ther.*, vol. 7, no. 3, pp. 3–5, 2017.

[31] A. M. Bonnick, M. Nalbandian, and M. S. Siewe, "Technological advances in nontraditional orthodontics," *Dent. Clin. North Am.*, vol. 55, no. 3, pp. 571–584, 2011.

[32] W. Orthuber and H. Fisher-Brandies, “Dental apparatus for bending and twisting wire pieces,” *United States Patent*, 1987.

[33] F. Riemeier, W. Butscher, F. Witte, C. Radinger, A. Cordell, and R. Sachdeva, “Machine vision system for apparatus for customized shaping of orthodontic archwires and other medical devices,” *European Patent*, 2014.

[34] A. Jacobson, “Department of reviews and abstracts,” *Am. J. Orthod. Dentofac. Orthop.*, vol. 107, no. 2, pp. 223–227, 1995.

[35] R. Sachdeva, “SureSmile Technology in a patient-centered practice,” *J. Clin. Orthod.*, vol. XXXV, no. 4, pp. 245–253, 2001.

[36] W. Butscher, F. Riemeier, R. Rubbert, and T. Weise, “Robot and method for bending othodontic archwires and other medical devices,” *United States Patent*, 2003.

[37] W. Butscher, F. Riemeier, R. Rubbert, T. Weise, and R. Sachdeva, “Robot and method for bending orthodontic archwires and other medical devices,” *United States Patent* , 2004a.

[38] W. Butscher, F. Riemeier, R. Rubbert, T. Weise, and R. Sachdeva, “Robot and method for bending orthodontic archwires and other medical devices,” *United States Patent*, 2004b.

[39] W. Butscher, F. Riemeier, R. Rubbert, T. Weise, and R. Sachdeva, “Robot and method for bending orthodontic archwires and other medical devices,” *United States Patent*, 2005.

[40] W. Butscher, F. Riemeier, R. Rubbert, T. Weise, and R. Sachdeva, “Robot and method for bending orthodontic archwires and other medical devices,” *United States Patent*, 2006.

[41] W. Butscher, F. Riemeier, R. Rubbert, T. Weise, and R. Sachdeva, “Robot and method for bending orthodontic archwires and other medical devices,” *United States Patent*, 2007.

[42] R. P. Scholz and R. C. L. Sachdeva, “Interview with an innovator: SureSmile chief clinical officer Rohit C. L. Sachdeva,” *Am. J. Orthod. Dentofac. Orthop.*, vol. 138, no. 2, pp. 231–238, 2010.

[43] A. K. Saxe, L. J. Louie, and J. Mah, “Efficiency and Effectiveness,” *World J. Orthod.*, vol. 34, no. 4, pp. 193–197, 2000.

[44] C. Schiemann, “Bracket system and method for planning and producing a bracket system for the correction of tooth malpositions,” 2016.

[45] S. Erich, “Orthorobot Lab Technology.” [online]: http://www.orthorobot.com/en/i-am-a-doctor/orthorobot-lab-technology.

[46] R. Rubbert and T. Weise, “Method and device for shaping an orthodontic archwire,”, *United States Patent*, 2010.

[47] R. Rubbert and T. Weise, “Method and device for shaping an orthodontic archwire,” *United States Patent*, 2007.

[48] C. A. Andreiko and M. A. Payne, “Custom orthodontic archwire forming method and apparatus,” *United States Patent*, 1995.

[49] J. Jiang, Y. Zhang, M. Jin, and C. Wei, “Bending process analysis and structure design of orthodontic archwire bending robot,” *Int. J. Smart Home*, vol. 7, no. 5, pp. 345–352, 2013.

[50] Y. Zhang and J. Jiang, “Analysis and experimentation of the robotic system for archwire bending,” *Appl. Mech. Mater.*, vol. 121–126, no. ISSN: 1662-7482, pp. 3805–3809, 2012.

[51] C. A. Andreiko and M. A. Payne, “Custom orthodontic appliance forming method and apparatus,” *United States Patent*, 2000.

[52] J. Jiang, Y. Zhang, C. Wei, T. He, and Y. Liu, “A review on robot in prosthodontics and orthodontics,” *Adv. Mech. Eng.*, vol. 7, no. 1, pp. 1–11, 2014.

[53] J. S. M. Vergeest, S. Spanjaard, and I. Horvath, “A methodology for reusing freeform shape content,” in *ASME Design Engineering Tecnical Conference & Computers and Information in Engineering Conference*, 2001, pp. 1–8.

[54] N. Al Mortadi, D. Eggbeer, J. Lewis, and R. J. Williams, “CAD/CAM/AM applications in the manufacture of dental appliances,” *Am. J. Orthod. Dentofac. Orthop.*, vol. 142, no. 5, pp. 727–733, 2012.

[55] A. Hatna and G. R.J, “An intuitive approach for the segmentation of parametric curves and cross-sections,” in *7th ISPE International Conference on Concurrent Engineering: Research and Applications*, 2000, pp. 473–480.

[56] J. R. Duflou, J. Váncza, and R. Aerens, “Computer aided process planning for sheet metal bending: a state of the art,” *Comput. Ind.*, vol. 56, no. 7, pp. 747–771, 2005.

[57] N. E. Smith and S. G. Strait, “PaleoView3D: From specimen to online digital model,” *Palaeontol. Electron.*, vol. 11, no. 2, p. 11A, 2008.

[58] Makerbot, “User manual,” 2013.

[59] O. Al-jubouri and A. Azari, “An introduction to dental digitizers in dentistry : systematic review,” *J. Chem. Pharm. Res.*, vol. 7, no. 8, pp. 10–20, 2015.

[60] I. B. Janko Hodolic, Tatjana Puskar, “Current status and future trends in dental cam restorative systems,” in *34th International Conference on Production Engineering*, 2011.

[61] I. Budak, D. Vukelic, D. Bracun, J. Hodolic, and M. Sokovic, “Pre-processing of point-data from contact and optical 3D digitization sensors,” *Sensors*, vol. 12, no. 1, pp. 1100–1126, 2012.

[62] T., Babak "Click3dPoint" (http://www.mathworks.com/matlabcentral/fileexchange/7594), Matlab Central File Exchange, Retrieved April 2017.

[63] T. Ursell, “Active Figure Zoom for Selecting Points.” https://www.mathworks.com/matlabcentral/fileexchange/ 38959), Matlab Central File Exchange, (2013), Retrieved May 2017.

[64] E. A. Nasr and A. Kamrani, “IGES standard protocol for feature recognition CAD system,” in *Rapid Prototyping: Theory and Practice*, A. Kamrani and E. A. Nasr, Eds. Boston, MA: Springer US, 2006, pp. 25–62.

[65] A. Mousali, “Convert form XYZ coordinate to YBC format.” (http://www.mathworks.com/matlabcentral/fileexchange/6100), Matlab Central File Exchange, Retrieved January 2017.

[66] S. Kapila and R. Sachdeva, “Mechanical properties and clinical applications of orthodontic wires.,” *Am. J. Orthod. Dentofac. Orthop.*, vol. 96, no. 2, pp. 100–9, 1989.

[67] J. Jin-gang, W. Zhao, Z. Yong-de, J. Ji-xiong, N. Suo-liang, and L. Yi, “Study on

springabck properties of different orthodontic wires in archwire bending process," *Int. J. Control Autom.*, vol. 7, no. 12, pp. 283–290, 2014.

[68] Y. Zhang, J. Jiang, and Y. Liu, "Theoretical analysis and experimental study of springback mechanism of archwire bending," vol. 6, no. 13, pp. 2495–2501, 2013.

[69] J. bong Yang, B. hee Jeon, and S. I. Oh, "The tube bending technology of a hydroforming process for an automotive part," *J. Mater. Process. Technol.*, vol. 111, no. 1–3, pp. 175–181, 2001.

[70] J. J. Y. Han and Y. Z. Y. Liu, "Springback mechanism analysis and experiments on robotic bending of rectangular orthodontic archwire," *Chinese J. Mech. Eng.*, 2017.

[71] J. P. T. Mo, F. Al-ayid, and T. G. Z. Chen, "Work planning for CNC multiple-bend pipe bending in small batch manufacturing," *Asian Int. J. Sci. Technol. Prod. Manuf.*, vol. 1, no. 2, pp. 77–87, 2008.

[72] D. Hao, W. Shaokui, X. Zeyang, G. Yangzhou, and X. Jing, "A Compensation method of springback effect for automatic orthodontic archwire preparation," *J. Integr. Technol.*, vol. 5, no. 3, pp. 20–27, 2016.

[73] L. C. Richards, G. C. Townsend, T. Bworn, and V. B. Burgess, "Dental arch morphology in south australian twins," *Arch. Oral Biol.*, vol. 35, no. 12, pp. 983–989, 1990.

[74] L. Lombardo, L. Saba, G. Scuzzo, K. Takemoto, L. Oteo, J. C. Palma, and G. Siciliani, "A new concept of anatomic lingual arch form," *Am. J. Orthod. Dentofac. Orthop.*, vol. 138, no. 3, p. 260.e1-260.e13, 2010.

[75] H. Noroozi, T. H. Nik, and R. Saeeda, "The dental arch form revisited," *Angle Orthod.*, vol. 71, no. 5, pp. 386–389, 2001.

[76] S. Lankton and A. Tannenbaum, "Localizing Region-Based Active Contours," *IEEE Trans. IMAGE Process.*, vol. 17, no. 11, pp. 2029–2039, 2008.

[77] R. M. Ricketss, "A detailed consideration of the line of occlusion," *Angle Orthod. Oct.*, vol. 48, no. 4, p. 274–282., 1978.

[78] R. H. W. Strang, "Factors of influence in producing a stable result in the treatment of malocclusion," *Am. J. Orthod. Oral Surg.*, vol. 32, no. 6, pp. 313–332, 1946.

[79] T. Triviño, D. F. Siqueira, and M. A. Scanavini, "A new concept of mandibular dental arch forms with normal occlusion," *Am. J. Orthod. Dentofac. Orthop.*, vol. 133, no. 1, pp. 20–26, 2008.

[80] K. H. Lu, "An orthogonal analysis of the form, symmetry and asymmetry of the dental arch," *Arch. Oral Biol.*, vol. 11, no. 11, pp. 1057–1069, 1966.

[81] S. Alharbi, E. A. Alkofide, and A. Almadi, "Mathematical analyses of dental arch curvature in normal occlusion," *Angle Orthod.*, vol. 78, no. 2, pp. 281–287, 2008.

[82] A. Muhammad, "Mathematical Analysis of Dental Arch of Children in Normal Occlusion : A Literature Review," *Int. J. Med. Dent.*, vol. 2, no. 1, pp. 33–40, 2012.