# A Data Physicalization Pipeline Enhanced with Augmented Reality

by

Doğacan Bilgili

Submitted to the Graduate School of Engineering and Natural Sciences

in partial fulfillment of the requirements for the degree of

Master of Science

Sabancı University

March, 2017

A DATA PHYSICALIZATION PIPELINE ENHANCED WITH AUGMENTED REALITY

APPROVED BY:

Assoc. Prof. Selim Saffet BALCISOY ..............................................
(Thesis Advisor)

Assoc. Prof. Bahattin KOÇ ..............................................

Asst. Prof. Sema ALAÇAM ..............................................

DATE OF APPROVAL: 07.03.2017

## Acknowledgements

I would like to start off by expressing gratitude to my thesis advisor Selim Balcısoy and all BAVLAB members for their support and feedbacks throughout the progress of my research.

I am pleased and feel thankful that Sema Alaçam and Bahattin Koç agreed to be in the jury for approving my thesis work.

Those who touched my life, they deserve an appreciation. My good-old roommates Mert Erpam, Can Özkan, Batuhan Arslan and Batuhan (Kemal) Yalçın, we had the most fun; My dearest friends Sera Giz Özel, Betül Avcı and Burcu Alptekin, I have always felt their support along the way.

I also would like to thank Pamir Mundt for providing me 3D printing service and Barış Dervent for helping me with photographing the tangible objects for the thesis.

Last but not least, my family; My mother, father, sister and grandma. Each of them made me the one I am today. I deeply feel grateful for their endless support and love throughout my life experience.

I would like to dedicate this thesis to future me and the one will stand by me.

A Data Physicalization Pipeline Enhanced with Augmented Reality

Doğacan Bilgili

Mechatronics Engineering, Master's Thesis, 2017

Thesis Supervisor: Selim Saffet BALCISOY

**Keywords:** Data visualization, Physical visualization, Augmented visualization, Augmented Reality, Tangible interfaces, Digital fabrication

## Abstract

Data visualization is an indispensable methodology for interpretation of information. The key purpose of traditional data visualization methods is to convert observed records into meaningful visuals to ease the cognition of trends. This virtual, passive technique on a display offers flexibility to create wide range of different visualization designs utilizing, however, only visual perception. Physical visualizations, on the other hand, enable sensations other than mere visual input, thus enhancing the experience and the impact. Although physical visualizations have some certain proven benefits over traditional visualizations, generating them is not as effective and quick. In that regard, need of physical models shaped around well-defined design rules are a prerequisite. Moreover, digital construction of the designed solid models for manufacturing is the next step to be achieved. However, even for a small set of data, constructing several models becomes a discouraging and highly time consuming task. This main problem is covered in this thesis by the implementation of an authoring tool. The introduced tool alleviates the burden of physical model generation process. Predefined models under design rules are generated in accordance with both the data input and adjusted parameters by the user. Utilization of digital fabrication techniques that are nowadays becoming widespread and easy to access is the key for physicalization. In order for an "Overview first, detail on demand" approach, an augmented reality tool is also introduced to work with designed

models so as to retain the physicality while presenting more detailed information such as exact values of data points along with augmented graphics if desired.

Arttırılmış Gerçeklik ile İyileştirimiş Veri Fizikselleştirme Sistemi

Doğacan Bilgili

Mekatronik Mühendisliği, Yüksek Lisans Tezi, 2017

Tez danışmanı: Selim Saffet BALCISOY

**Anahtar Kelimeler:** Veri görselleştirme, Fiziksel görselleştirme, Arttırılmış görselleştirme, Arttırılmış Gerçeklik, Taşınabilir arayüzler, Dijital üretim

# Özet

Bilginin yorumlanması söz konusu olduğunda veri görselleştirme vazgeçilemez bir yöntem olarak ortaya çıkmakta. Bilinen veri görselleştirme yöntemlerinin asıl amacı, kayıt edilmiş gözlemlerin anlamlı görsellere dönüştürülerek algılanmasının kolaylaştırmaktır. Bir ekran yardımı ile gerçekleştirilen bu pasif sanal görselleştirme yöntemi her ne kadar çeşitlilik sağlamak konusunda esneklik sunsa dahi sadece görsel algımızı kullanmamıza izin vermektedir. Fakat fiziksel görselleştirmeler görsel algımıza ek olarak diğer duyularımızı kullanmamızı da sağlayarak daha etkili bir deneyim yaratmaktadır. Süregelen görselleştirme yöntemlerine kıyasla fiziksel görselleştirmelerin kanıtlanmış faydaları bulunsa da, bu fiziksel görselleştirmeleri gerçekleştirmek diğerleri kadar verimli ve hızlı bir şekilde yapılamamaktadır. Bu bağlamda, iyi belirlenmiş tasarım kuralları çerçevesinde oluşturulmuş fiziksel modellerin oluşturulması bir ön koşul olmaktadır. Bunun ardından gerçekleştirilmesi gerekli olan ikinci adım ise üretime hazır katı modellerin oluşturulmasıdır. Ancak küçük bir veri seti için dahi bir kaç tane fiziksel veri modeli oluşturmak oldukça yıldırıcı ve vakit alıcı bir işe dönüşmektedir. Bu ana sorunsal göz önünde bulundurulup tez konusu olarak bir yazılım geliştirilmiştir. Tez dahilinde sunulan yazılım fiziksel model oluşturma sürecindeki zorlukları ortadan kaldırmak amacını taşımaktadır. Kullanıcı tarafından belirlenen değişkenler ve kullanılmak istenilen veri dosyasına göre daha önceden belirlenmiş kurallar çerçevesinde oluşturulan mod-

eller bu yazılım tarafından üretilmektedir. Günümüzde giderek yaygınlaşan ve erişilmesi kolaylaşan dijital üretim yöntemlerinin kullanılması, veri fizikselleştirme konusunda önemli bir rol oynamaktadır. 'Önce genel hatlar, talep edilirse detay' düşüncesi ana fikir olarak alınıp, gerek duyulduğunda daha fazla detayı fizikselikten ödün vermeden sunabilmek için, üretilen fiziksel modeller ile çalışan bir arttırılmış gerçeklik yazılımı da ek olarak geliştirilmiştir.

# Table of Contents

# List of Figures

# List of Tables

## Chapter 1

# Introduction

The history of visualization dates back to pre-historical time periods as a medium for story telling and conveying information. The urge to use visuals for communication purposes is intrinsic to human nature. Some of the most potent evidences of this are cave paintings as the very first visuals created by the human kind. Depicted visuals on cave walls serve in various ways, from telling stories depending on what is observed to keeping records of hunting activities. Later, with the emergence of ancient civilizations, this primitive practice evolves into more advanced and systematic methods such as pictograms that are able to exhibit more detailed concepts [7]. These examples reveal the fact that the use of visuals from very early ages, all along, has been the origins of visualization, which formed the rudimentary stages of various artistic, scientific and statisticial disciplines such as cartography, astronomy and statistical graphics [1]. As with all disciplines, data visualization is in an endless refinement process. Development of fields like psychology, computer science and graphics effectively contribute to data visualization [7]. The most prominent advancement can be considered to be the emergence of data graphics in the 19th Century [8] and then with the introduction of computerization methods, generation process of data graphics becomes an effortless operation and that results in field of statistical graphics [1, 9, 10]. On top of the concise introduction, this chapter covers and provides an overview for the short history of data visualization and its key implications along with major examples in section 1.1. Then the scope becomes more specific and narrows down from general data visualization to ancient examples of data physicalization in section 1.2.

## 1.1 Short History of Visualization

A rug plot in figure 1.1 by Friendly [1] illustrates the density trend of visualization over centuries. As maintained by Friendly, history of visualization is divided into eight distinct eras over the course of five centuries. When the overall fashion is considered, a growing trend is observable until the end of the 19th Century and then a plunge occurs, which lasts until the mid-20th Century, specified as the "Modern Dark Ages", followed, again, by a rise.

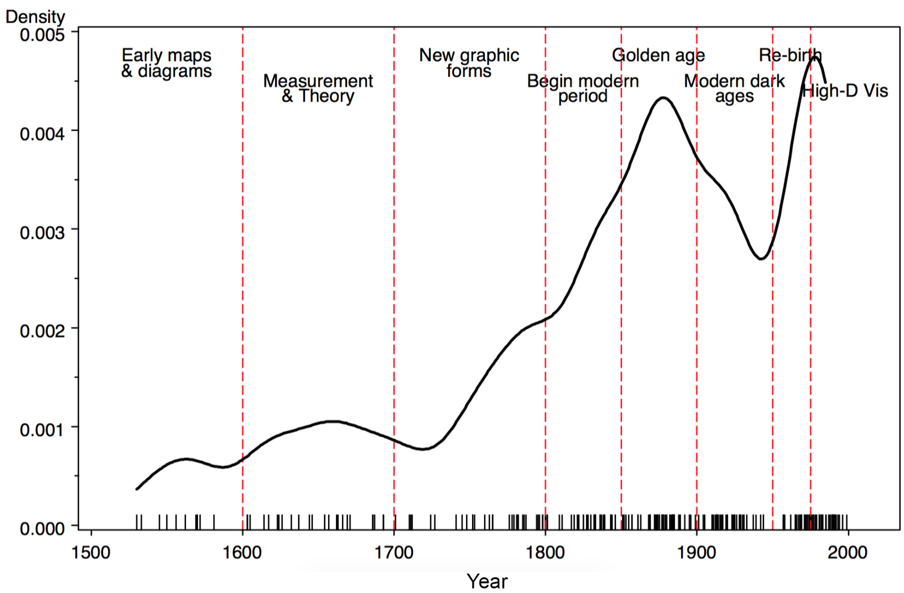


Figure 1.1: Time course of developments in visualization illustrated by Friendly [1]

Pre-17th Century paved the way for quantitative data and statistical graphics. Although the primary concepts were not well-established before the 17th Century due to lack of scientific methods, various examples suggesting these elementary notions emerged [10]. Most of those early examples were in the field of cartography and astronomy, for purposes such as providing map solutions in order to ease the positioning problems in sailing with further assistance [11]. Astronomical implications appear as list of tables for keeping results of heavenly body observations [1]. Examples from Egyptian surveyors to depict their lands with illustrations and by the same token picturing the courses of celestial bodies on a grid system by an unknown observer in the 10th Century, shown in figure 1.2, were suggesting the very first examples of coordinate systems along with use of quantitative information changing over time [2, 5]. Towards the 16th Century, further advancements in instrumentations and scientific methodologies led to the subject of data visualization [1].

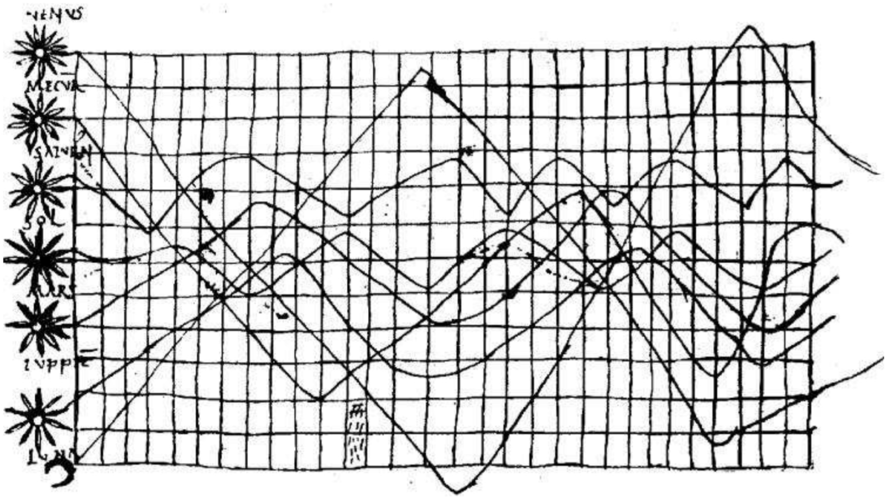

Figure 1.2: Courses of observed celestial bodies from the 10th Century

The most noteworthy, contribution along the way was the introduction of Cartesian coordinate system by the French philosopher and mathematician Ren Descartes in the early 17th Century. This principle proposed the method of specifying points with numeric values in space with respect to a reference origin. The consequence of the foundation was introduction of a systematic grid, which would later be utilized by most of the well-defined visualization methods [1]. Furthermore, in that time period, a work by Michael van Langren, shown in figure 1.3, constitutes the use of the first statistical data for visualization [12]. This example, depicts 12 unique estimations for the distance difference in longitude between Rome and Toledo cities. The importance of this visualization is the use of one dimensional scale to show the estimations relative to a reference point rather than providing a tabular display for the estimated data by various astronomers. That makes Van Langren's work to be considered as the first use of effect ordering for data displays [13].

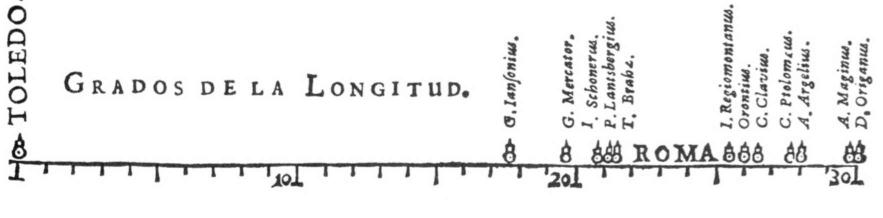


Figure 1.3: First statistical data visualization from 1644

One notable visualization of an astronomical observation from the early 17th Century, belonging to Galileo Galilei [14], is shown in figure 1.4. Illustrations from the summer of 1612 depict the phases of the Sun spots over a period of time. The significance of this set of visualizations is its time variant nature, meaning the change of visuals as a function of time.

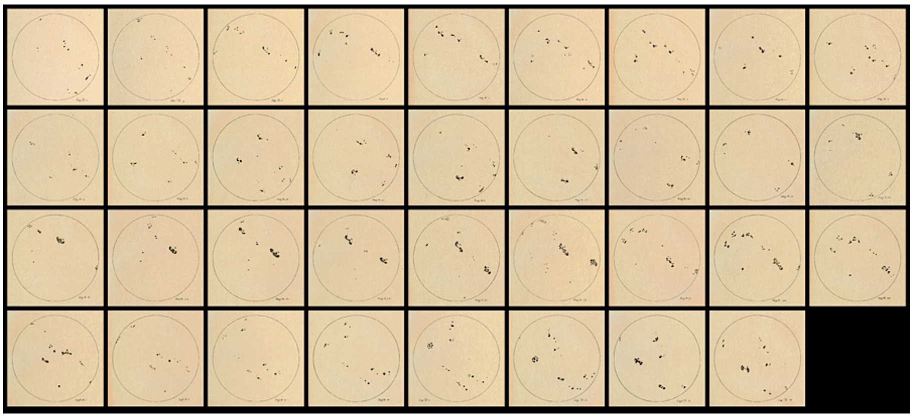

Figure 1.4: Sunspot visualizations of Galileo

In the 18th Century, knowledge and techniques were mature enough for further developments in graphical visualization. Systematic collection of empirical data and introduction of methodologies such as interpolation to form new data points from discrete set of known points were some of the more notable contributions in this century [1]. One of the most remarkable achievements bridging the 17th and 18th Century was William Playfair's invention of line graph, bar chart, pie chart and circle graph, which are still widely in use as modern visualization methods [15–18]. One example of Playfair's visualizations depicting the records of all the imports and exports to and from England between the years of 1700 and 1782 is shown in figure 1.5.

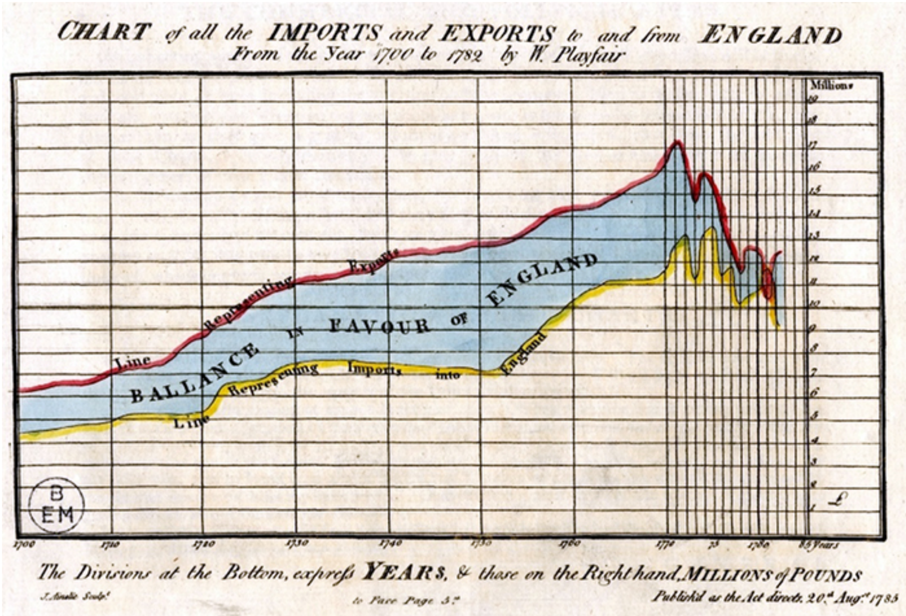


Figure 1.5: Visualization for all the Imports and Exports to and from England By William Playfair in 1785

Towards the modern ages of data visualization, quite a few contributive works by a number of diverse people was made. Since then, names such as Jacques Bertin, John Tukey and Edward Tufte appeared as the modern contributors. "The Semiology of Graphics" by Bertin compiles an extensive study focusing on different features of geometries, titled as "Retinal Variables". In his work, Bertin introduces some rules for principles of graphic communication and defines systematic organization for legit visuals construction [19].

Nowadays, mathematical theories and computer graphics are extremely well-developed for processing large amounts of data and visualizing them through various computer software with ease. On top of that, methods and methodologies for data collection are also effortless thanks to advancements in both software and sensor technologies [9]. However, the way we interact with data through visualizations has not changed so much since then. 2-Dimensional (2D) and 3-Dimensional (3D) on-screen visualizations utilize only the perception of vision on the 2D surface of a screen or paper. Although touch screen devices augment the interaction and introduce the tactile sense to a certain extent with several hand gestures, this method of interaction does not deploy the full capabilities of hand, such as grasping, manipulation and texture/material sensation [20]. At this point, physicalization of data by encoding information into physical objects emerges as a method for expanding the interaction techniques, thus deploying additional senses. As a recently-developed research area, physicalization appears as a subset of human-computer interaction (HCI). HCI, as the name implies, studies and designs interface solutions for new interaction technologies between human and computer. Increase in complexity and size of the data of interest prompts more interactivity in the designs of new visualization techniques over static and interactively-limited displays with WIMP (Windows, Icons, Menus, and a Pointer) interfaces. HCI is offering myriads of interaction techniques, whereas attention on interaction techniques specifically for information visualization (infoVis) is quite rare compared to other topics. Having said that, this gap between HCI research and infoVis techniques is recently closing with further focus on attempts to bridge two fields [21].

In the following section, very early examples of data physicalization in various forms from pre-historical time periods will be examined and their features will be briefly discussed. This will be followed by a literature review of data physicalization in Chapter 2.

## 1.2 Early Examples of Physicalization

Examples of information embedded into physical objects appear as a natural practice in the ages when writing and languages were absent. Found artifacts show signs of abstraction of information in various formats. One example shown in figure 1.6 is the Blombos ocher plaque, which is believed to be approximately 70,000 - 80,000-years old [22]. The signs on the plaques are considered to be a systematic pattern and therefore suggest the idea of information storage purpose according to some researchers, although it is not completely certain that this artifact encodes a specific piece of information. There are several interpretations for the potential meaning of the signs, categorized as 'numerical', 'functional', 'cognitive', and 'social' by Cain [23].

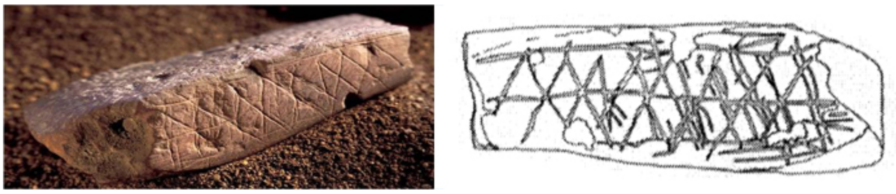

Figure 1.6: Blombos ocher plaque from the Middle Stone Age

Another artifact named the Lebombo bone, has been dated to 35,000 B.C, is considered to be an object for keeping the record of lunar cycles, therefore regarded as the first mathematical artifact. Figure 1.7 depicts the Lebombo bone, which is consists of 29 distinct notches deliberately carved onto a baboon fibula. The significance of the number of notches is the potential link between the number of lunar cycles, which is approximately 29.531 days. Besides, use of an advanced counting system suggests the signs to be the birth of calculation [24].

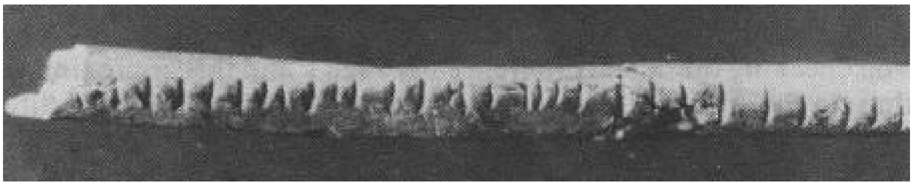

Figure 1.7: The Lebombo bone found in the Lebombo mountains of Swaziland

Yet another baboon fibula found in Ishango named the Ishango Bone, is a 20,000-year-old mathematical artifact, shown in figure 1.8. Similar to the Lebombo bone, it also has notches on it but unlike the Lebombo bone, there are three different columns of notches categorized as the middle, left and right sections. Since the actual function of the bone is an enigma, various assumptions drawing the inference that the bone might be indicating the knowledge of basic arithmetic or used as a lunar calendar [25]. Latter is supported by the fact that some African civilizations are still utilizing bones and various similar objects as calendars.

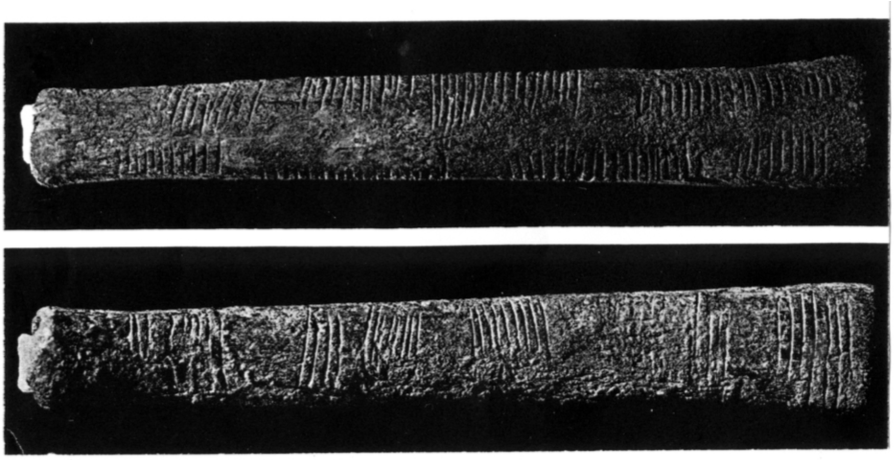

Figure 1.8: The Ishango bone dated to the upper Paleolithic era

Clay tokens appearing around 8,000 BC before the invention of writing, as a precursor. The primary function of those tokens is to reckoning the amount of physical entities. Each token in various shapes symbolizes an object and keeps record of the amount of a specific item, which is emerged from a need for agricultural settlement, since nomadic communities were relying on an egalitarian system [20]. With settlement, formation of states prompted the necessity of keeping better track of commodities due to the emergence of personal property concept, which consequently led to trading and bartering systems. Use of a token system is an evidence of the "Concrete" counting, as any numeric abstraction system was to be developed at that time. Varying token shapes made it easier to count distinct items by establishing a link between the item and the token through assignment, which is also known as the "one-to-one correspondence" system [26]. The significance of clay tokens can be regarded as the introduction of an early method for object counting and an act of embedding information into a physical object.

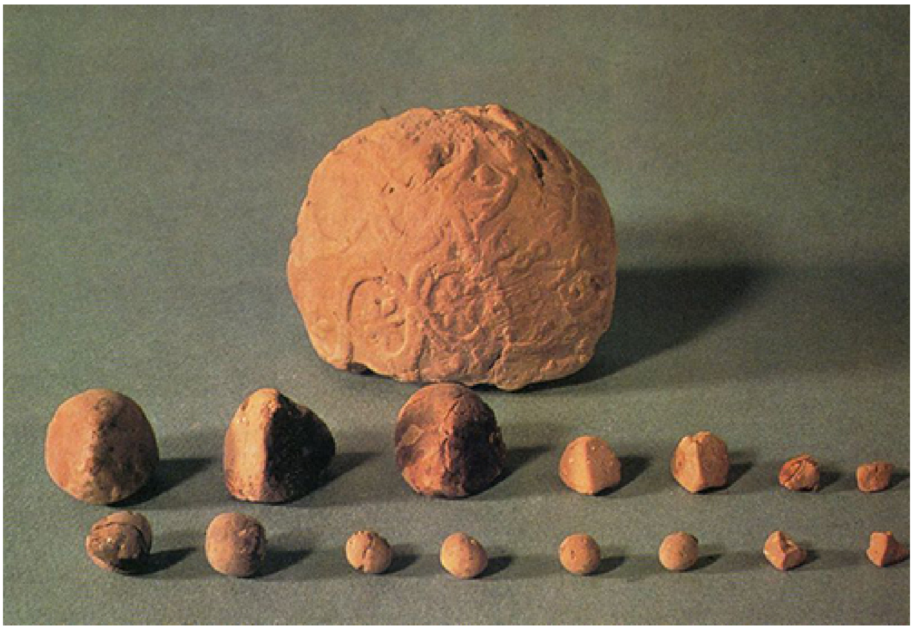

Figure 1.9: Clay tokens with an envelope

One relatively sophisticated example compared to the previous ones is quipu, shown in figure 1.10. It appeared around 3,000 BC and was utilized until the 17th Century. Quipu is a set of threads with various features such as the color of threads, number and type of knobs attached to a chord. It is mostly used for encrypting information, which depends on the relative position, number and type of knobs on each thread. This physical model is believed to be used for accounting purposes. Since the complexity of the method of information encoding is high with quipu, it was unlikely that anyone could interpret the information but the creator of the specific quipu.

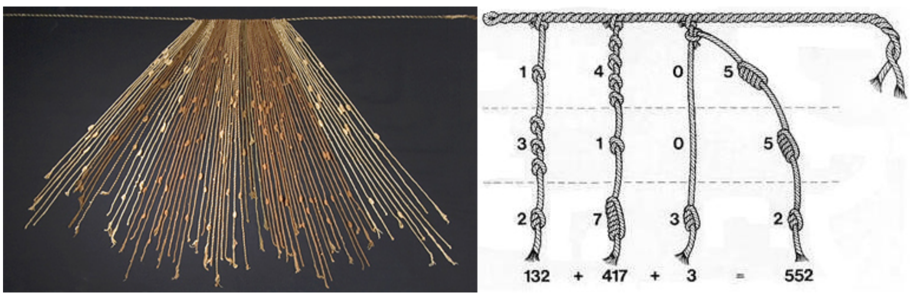


Figure 1.10: A quipu and an example of encoding

In the following chapter, detailed review of contemporary literature content for data physicalization is presented.

## Chapter 2

# Literature Review

Modern literature of data physicalization, as a recently developing research topic, is relatively limited in terms of number of unique work. Having said that, available publications cover various aspects of the research topic such as cognition, design and manufacturing techniques to demonstrate comprehensive outputs. Case and design studies are the primary methods for revealing the effectiveness of physical visualizations.

In [2], the importance and impact of digital fabrication techniques and their implications on data physicalization are discussed. The problems, which are related to design and manufacturing of physical visualizations and considered to be unsolved, are also explained. Those problems are listed as *Manufacturability*, *Assembly & Fit*, *Balance & Stability* and *Strength*. On top of that, a case study illustrates the need of an authoring tool for data physicalization and this is followed by the introduction of the tool named MakerVis, shown in figure 2.1, and design sessions for surveying the interaction of users with the software. Depending on the reviewed literature and the claim of the authors, this tool is considered to be the first authoring tool for facilitating the data physicalization processes [27].

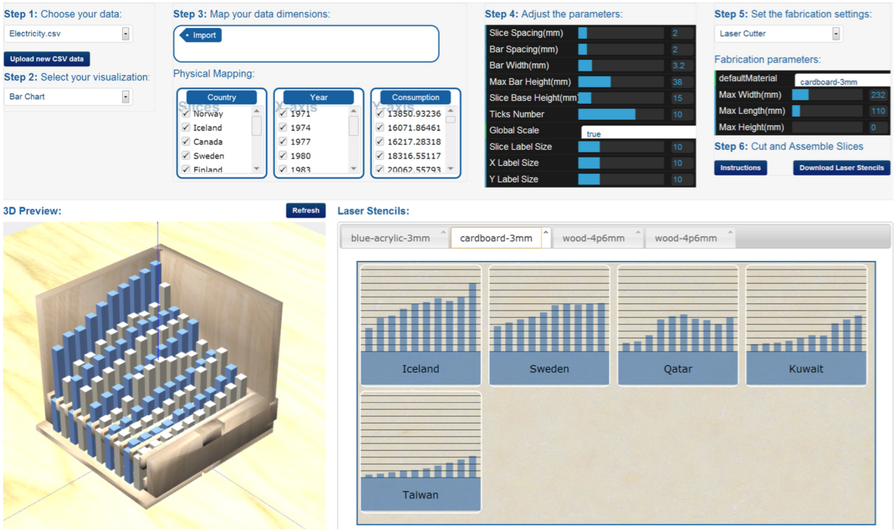


Figure 2.1: Interface of MakerVis introduced in [2]

The effectiveness of visualizations is another concern subjected to research. There are various studies conducted for analyzing the possible potential of physicalization. Jansen et al. [3] introduces the first information visualization (infovis) study for comparison of traditional on-screen visualizations with 3D physical visualizations, considering the fact that 3D visualizations are regarded to be problematic on 2D screens. Findings of the study reveal that 3D physical bar chart visualization performs superior compared to on-screen equivalence in terms of information recovery with the help of sense of touch, whereas the role of manipulability was found to be less supportive. Furthermore, this study suggests that the success of passive physical visualization unveils the feasibility of dynamic physical visualizations as more interaction and variation is viable with them.

Another research done by Gwilt et al. [28] investigates the possible help of physical data objects for proper communication between cross-sectors. Statistical data, which is presumed to be strenuous for many people without scientific background, is used for a study to clarify the questions of whether physical objects change the cognition of the information embedded or aroused any new interpretations; If material qualities play any role for understanding the data. This study was conducted with three different groups of people with backgrounds of design, science/engineering, and people with none of those backgrounds. It is reported that the meaning of physical objects became more prominent for all study groups upon the explanation of original 2D visualizations. It is also noted that physical objects prompt people to start discussions around the data with less effort. Therefore, they are considered as an influential communicative tools by participants.

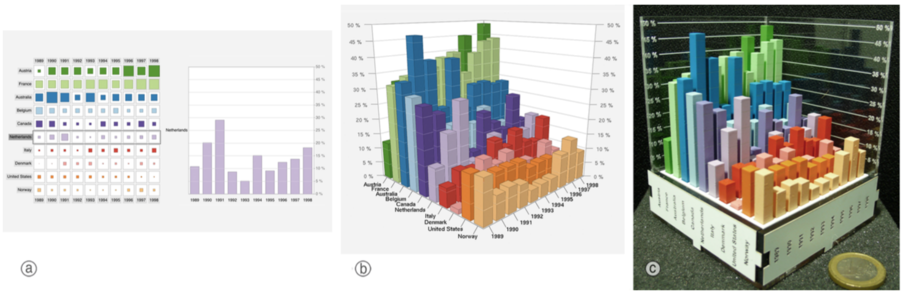

Figure 2.2: Three different types of visualizations used for the study in [3]. a) On-screen 2D visualization; b) On-screen 3D visualization: c) Physical 3D visualization

Memorability and being easy to be understood are some of the indications on level of effectiveness of a visualization. In that regard, potential of manipulation and capability to induce multiple sensations help physical visualizations to excel. The extensive study conducted by Borkin et al. [29] tries to find any intrinsic feature affecting the memorability of visualizations given the proven fact that certain images are more memorable due to their innate characteristics [30]. It is revealed that ordinary visualization types compared to distinctive models are less memorable and various attributes such as color and being easily distinguishable contribute to increase of memorability. By Kosslyn and Cleveland [31, 32] it is noted that plain and clear visualizations are easy to understand, thus to be remembered, unlike chart junk. Stusak et al. [33] conducted a study with 2D and 3D bar charts for measuring the success rate on recall of information with immediate and delayed timing. Physical properties, such as weight, volume, texture are considered when 3D models are concerned. Results showed that 3D visualizations perform better than 2D on recall of information for one data set. Other data set did not show any prominent variance. This conclusion prompted to claim that spatiality and tangibility performs as expected when the data set is engaging. Otherwise, no enhancing effect on recall performance is observable. Another study by Stusak et al. [34] displays a result that after two weeks time period, recall of information with 3D visualizations are remarkably higher specifically for maximum and minimum points in data sets. Marshall [35] suggests that tangible interfaces encourage learning through natural interaction, compared to WIMP interfaces. Increase of accessibility to a wider range of people as a consequence of intuitive form of physical objects is another engaging feature. Furthermore, when collaborative learning matters, physical objects are found to be appropriate with their interactive quality. Since tangible interfaces enable more senses, urge of discovery enhances the understanding of the intended information. Khot et al. [4] conducted a study for observing the effects of physical artifacts on daily physical activities of people. A software was provided and 3D printing facility was deployed in houses of six different participant groups. Five different physical models, shown in figure 2.3, provided for visualizing various derivations of heart rate data. Results yielded that a physical representation of daily activity data attracts more attention and raises more awareness compared to traditional on-screen visualization of heart beat rate. A tangible object received upon a physical activity acts like a reward and motivates the recipient as an indication of Goal-setting theory [36]. A similar study done by Stusak et al. [37] for investigating the long term influence of data physicalization on running activity. Three weeks long examination inspected influential results on participants. It is found that several aspects of tangible objects play a role in behavioral change of participants related to running activity. Physical models are handy and therefore always easy to access in comparison to on-screen visualizations. Having a physical object, representing the data of each running activity, prompts participants to compare previous models with recent one. Furthermore, by altering the running habits, participants attempt to shape their physical models. All those aspects found to be supportive for motivation and enhance the commitment to activity.

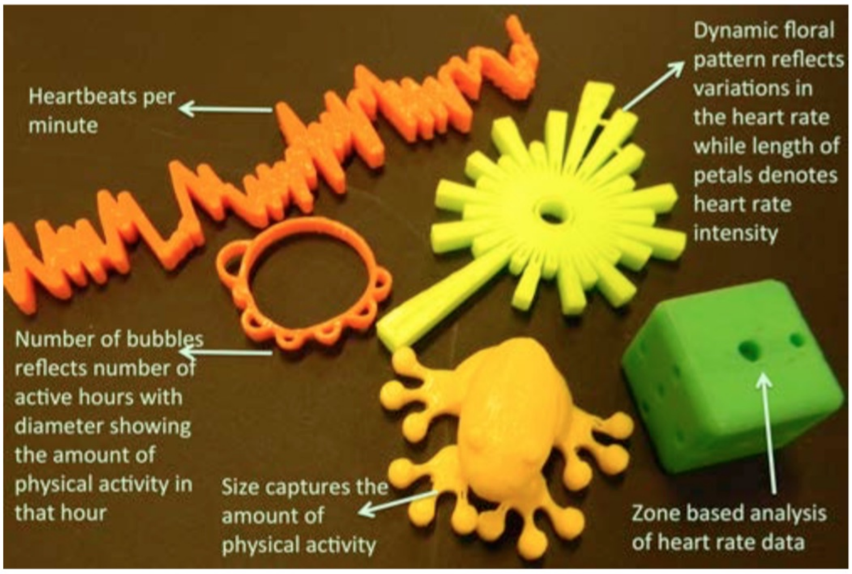


Figure 2.3: Five different physical visualizations for physical activity data used in [4]

Stusak and Aslan [38] concluded that well-developed designs are critical for acceptable analytical tasks with physical visualizations. They presented an experimental work on physical visualization by an attempt to create original physical model designs with simple tools and conduct of a study to measure information retrieval performance of those new designs. Another study evaluating the feasibility of physical visualizations by Szigeti et al. [39] conducts an enquiry into interaction of users with physical visualization objects along with the impact of tangible objects in collaborative work.

Unlike previously discussed publications, Nadeau and Bailey [40] experiments physical visualization in a medical setting. Volume data sets of human skull and brain were converted into 3D interlocking models and printed with a high precision industrial standards 3D manufacturing machine. The motivation for physically visualizing volume data sets was manipulation of the 3D tangible visualization in a natural way and not requiring high-cost computer hardware for on-screen visualization. Being able to observe the interaction of interlocking parts was considered as another key aspect of physicalization in that use case.

# Chapter 3

# Thesis Motivation and Contribution

This chapter illustrates the motivation for the research topic and its contributions to literature. Section 3.1 gives insight into studied research topic and provides reasoning for the work carried out by laying the groundwork. Section 3.2 gives grounds for the novelty of the thesis work by presenting the contributions to what has been disregarded in literature.

## 3.1 Motivation

Analysis of a set of data mostly depends on the way it is interpreted. A set of numbers might not reflect the underlying meanings when not visualized in an appropriate way. When 'Big Data' is involved, significance of visualization becomes even more prominent due to increased complexity. Traditional on-screen visualization methods are well-developed since their emergence. Although these visualizations are comprehended by the people dealing with any type of data, their impact on telling the story of the data of interest becomes less intense and subsides eventually. This is due to the fact that human cognition is prone to be less impressed by what is already well-known. At this point, physical visualization subject manifests itself as an alternative to on-screen visualizations.

By their tangible nature, physical visualizations introduce new means for perception of the data. This is well-studied by various researchers and it is proven that physical data objects have certain significant benefits over on-screen visualizations with limited interactions as summarized in Chapter 2. What physical visualizations offer is beyond interaction capabilities of WIMP interfaces and therefore considered to be more interactive and engaging. However, this novel research topic was yet to be fully studied due to lack of advancements in additive manufacturing technologies, enabling fast and easy manufacturing of tangibles. Recently, evolution of those technologies peaked and it is further developing, consequently becoming more accessible. Accessibility in rapid prototyping technologies is crucial in terms of paving the way for researchers to conduct effortless study on data physicalization subject.

As illustrated by the reviewed literature, data physicalization as an emerging research topic is mostly studied in terms of its impact on cognition and how well it can augment the potential of on-screen visualizations. Another research question studied several times is whether physical visualizations can be a good substitute for on-screen visualizations. All those aspects of physical visualizations still need to be subjected to further investigation for rigorous grounds.

Although physical visualizations are recently becoming a popular research topic, software basis for generating tangible data visualization models are almost completely neglected. Process of generating physical models unique to the data of interest is an arduous procedure, therefore in the absence of such an authoring tool, practicality of data physicalization would quickly become a doubt. The literature features only one explicit study by Swaminathan et al. [27] regarding development of an authoring tool for physical data model generation for fabrication. Khot et al. [4] and Stusak et al. [37] mention use of such tools for generating physical visualization models, whereas explicit details on those tools are not available.

Despite the promising potential of data physicalization, in some cases they are liable to lack some features such as labeling for complete presentation of data depending on the model design. Having this problem as an encouraging reason aside, augmentation of physical models also enhances the capabilities by enabling introduction of further information and interactivity.

The motivation for this thesis work is closing the aforementioned gaps by developing a pipeline for data physicalization. The introduced authoring tool utilizes rapid prototyping technologies, specifically 3D printing and laser cutting, in order to automate the process. Development of an augmented reality interface working specifically with physical model outputs of the authoring tool is complementary to pipeline.

Contributions of the thesis work for addressing the aforementioned problems in data physicalization subject is detailed in the following section.

## 3.2 Contributions

In the context of this thesis work, following contributions are made;

- New physical model designs

  Unlike well-known on-screen visualization styles, there are no such standardized models for physical visualization due to the fact that the subject is immature. That being the case, the only authoring tool example introduced in [27] proposes limited number of models. In this thesis work, two new physical models are presented in the light of proven design rules.

- Introduction of an authoring tool for effortless and rapid physical model generation

  Given the fact that creating physical visualization models are significantly demanding both in terms of design and manufacturing, an authoring tool is developed for generating, displaying and outputting the physical models in accordance with a data input and specified parameters by a user. As the authoring tool eases the process of model generation, make use of rapid prototyping technologies reduces the time in demand and increases the accessibility. The significance of this contribution is the likelihood of catalyzing the potential use of tangible data models in daily life such as adoption as personal objects or collection as motivational tools that visualizing personal data.

- Augmented Reality interface for enhancing the physical visualization models

  Data physicalizations are unusual yet appealing through utilization of more sensory inputs and interactivity. Having said that, versatility of a on-screen visualization is hard to achieve with physical visualizations. This problem can be dealt through use of augmented reality to minimize the constraints of tangibles to a certain extent. An augmented reality interface is designed for each physical visualization model type to illustrate the practical use of AR with physical visualization objects.

## Chapter 4

# Preliminaries and Background Information

This chapter is preliminary to upcoming chapters. Required background information is provided and detailed explanations are made for a smooth reading throughout the thesis.

Enabling technologies are detailed along with used technologies for fabrication of implemented software outputs. This is followed by three specific file formats used within the developed software. Those file formats for data management, 3D printing and laser cutting are detailed and their use in the software is clarified. On top of that, supportive software and SDK packages used for implementation of the augmented reality application are presented along with further information on augmented reality itself. Finally, a concise information on the classification of data types is given to avoid any misconception.

## 4.1 Digital Fabrication

Enabling technologies are indispensable for the effortless fabrication of tangible objects without the need for any expertise and complex, beyond-reach processes. Advances in those technologies also provide opportunities and possibilities for the daily life feasibility of various research topics, including data physicalization, by enabling users to manufacture tangible object with ease. In this thesis the focus will be on 3D printing and laser cutting technologies as they are easy and low-cost to be obtained or serviced. Although those technologies are considered as rapid prototyping tools, their output can be used as a final product in data physicalization context unless their manufacturing capabilities are pushed beyond limits, which merely depends on the design of a physical model. Those rapid prototyping technologies can be divided into two subcategories as additive and subtractive techniques [41].

### 4.1.1 Additive

Additive fabrication technique is the description for process of creating cross-sections of a 3D object layer by layer. Although 3D printing is recently becoming popular and affordable in houses, its appearance dates back to the late 1980s [42,43]. There are several different technologies developed for 3D printing. Some of them are listed as follows;

- Stereolithography (SLA)
- Digital Light Processing (DLP)
- Fused deposition modeling (FDM)
- Selective Laser Sintering (SLS)
- Selective laser melting (SLM)
- Electronic Beam Melting (EBM)
- Laminated object manufacturing (LOM)

Different methods work with materials in different shapes. As SLA method requires a liquid resin container and a light source for solidifying the resin, SLS method works with a powder bed and a laser source, whereas FDM method heats a thermoplastic material for extrusion through a nozzle. Other systems nearly use similar methods with different technologies. As illustrated, there are various technologies developed for 3D printing. However, the most commonly used technology is FDM printers, which are widely available in the market as desktop printers, due to simplistic and low-cost technology they are relying on.

With an FDM printer, illustrated in figure 4.1, solid thermoplastic material becomes fluid upon heating and is extruded through a nozzle for construction of each cross sectional layer of a 3D object. Layers, built on top of each other, form a final 3D shape. Various variables such as step precision of actuators, diameter of the nozzle and the control algorithm have a direct effect on the resolution quality of the final product.

Until recently 3D printer technology was not accessible due to its high market price, which was listed around $45,000 in 2001. Nowadays, personal 3D printers have reasonable prices and highly affordable [44].

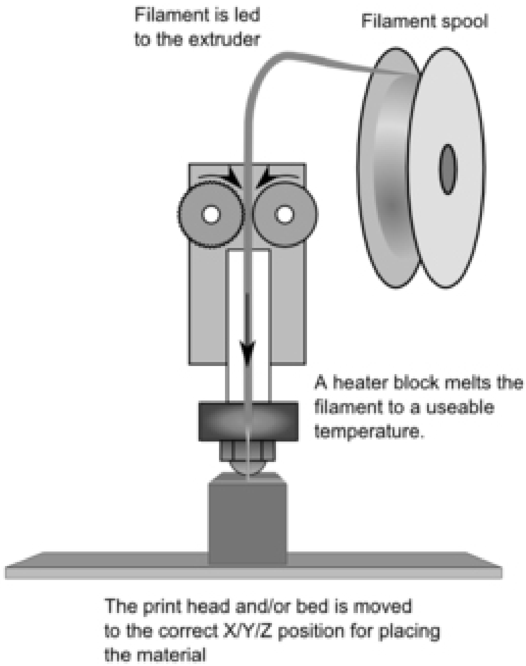


Figure 4.1: Working principle of FDM printers

### 4.1.2 Subtractive

In contrast to additive fabrication technique, working principle of subtractive fabrication depends on removal of tiny layers of material from a bulk to produce a desired geometry. Although subtractive manufacturing can be done manually, various technologies are available for high precision results. Computer Numeric Control (CNC) enables the automation of numerous tools with the intention of material removing. 3-axis CNC mills and laser cutters are the most affordable examples. As former uses a spindle for cutting away the material, latter method utilizes high power laser technology for cutting or engraving the material depending on the laser power. Despite their affordability, making use of those technologies as a personal tool requires extra care due to their working principle. By virtue of being subtractive, those techniques produce waste material as chips and smoke in case of laser cutting, consequently requiring ventilation and vacuuming [44].

## 4.2 File Formats

In this thesis work, three different file formats are utilized to be an input into and an output from the implemented software. Those formats are selected to be generic and cross-platform so that need of conversion is minimized, even eliminated.

### 4.2.1 CSV

CSV stands for 'Comma-separated Values'. As the name implies, each entry is separated by a comma to form a tabular structure. A line of entry separated with commas forms columns and each newline entry forms a row. The idea is straightforward, whereas the use of commas as a field separator may lead various issues, especially when the field contains a comma character. In order to address this problem, use of quotation marks to enclose the field is an option. However, this is not a complete solution. Use of valid delimiting characters are supported by parsing algorithms and may solve possible problems to a certain extend. These problems can be considered as a trade-off between simplistic structure and capabilities of CSV file format. As all data storing file formats, CSV requires a parsing algorithm to be resolved. Some parsers are capable of distinguishing between string and numeric entries and store the variables accordingly.

CSV is used as an input file format for the implemented software considering the likelihood of low complexity of a prospective dataset.

An example for the output structure of a CSV file is shown in table 4.1

| | | | | |
|---|---|---|---|---|
| FirstRow | Feature1 | Feature2 | Feature3 | Feature4 |
| SecondRow | 10 | 20 | 30 | 40 |
| ThirdRow | 50 | 60 | 70 | 80 |

Table 4.1: CSV file format example

### 4.2.2 STL

STL (StereoLithography) is a file format for defining surface geometries of a 3D model, created by 3D Systems company [45]. Within the file, vertices and normal vectors are listed to define each surface of a geometry. Therefore, the more detailed the geometry, the larger the file size. STL file does not define any units for the distances [46]. Unit of each length depends solely on software interpreting the STL file. STL is broadly used by wide range of Computer aided design (CAD) and 3D computer graphics programs and easy for interchanging geometry data between different environments. STL is the most common file format for 3D printer softwares, therefore considered to be universal. VRML (Virtual Reality Modeling Language) is an alternative format to STL, which also defines colors for specific geometries. VRML is used by 3D printer systems with multiple extruders for multi-color printings.

In the implemented software, STL file format is used for outputting 3D geometries to be printed. It should be noted that not every STL file is suitable for 3D printing. There are certain requirements to be met in order for a STL file to be printable. This is discussed further in Chapter 5.

### 4.2.3 DXF

DXF (Drawing Exchange Format) is a file format developed by Autodesk company for the purpose of storing and exchanging vector CAD drawings. Autodesk publishes documents explaining the syntax for the specifications of the DXF file format. The syntax is a set of rules that should be complied for a valid file generation. DXF operates with group codes ranging from 1 to 1071. Each code represents a specific feature. With the help of official documents published, one can generate a desired DXF file. Unlike STL file format, DXF can specify units for distances between points. It is a file format suitable for storing both 2D and 3D drawings data and is a universal file format for CNC machines. In the software developed for the thesis work, it is used for storing and outputting 2D drawings, which are meant to be manufactured through a laser cutting or a CNC milling machine.

## 4.3 Augmented Reality

Augmented reality (AR) is a technology for blending real-life environments with virtual graphics in real-time. As opposed to virtual reality (VR), which is generation of a complete virtual environment independent of reality, in AR, information is superimposed to environment which is being observed at the moment. This method supports enhancing the reality by introducing extra information with visuals. In that sense, AR is an immersive experience without compromising the reality.

Basic AR technology requires a marker to be tracked in order to define a reference point in the space for overlaying information on top of real-life environment accordingly. Fusion of additional sensors such as a camera for vision and an inertial measurement unit (IMU) for orientation sensing are required in order to exactly know where the observer is. Basically, an algorithm tracks the target, whose exact dimensions are specified, and real-time input from camera is fused with orientation input from IMU and desired graphics are displayed on top of the camera input with a correct size, position and orientation through a display.

AR applications are mostly demonstrated on tablet computers due to their widespread use and availability. However, there are also dedicated headsets developed specifically for AR applications. Head mounted display (HMD) and head-up display (HUD) are some of those head-set technologies. Microsoft HoloLens and Google Glass can be listed as well-known examples to those headset displays respectively. In contrast to the screen of a tablet computer, HDM and HUD devices reflect information on a transparent surface, which the user can see through. Figure 4.2 shows examples of AR on a tablet computer and a headset display.

There are several software development kits (SDK) working on various platforms for fast prototyping. Vuforia is the SDK used along with Unity game engine in this thesis work.


Figure 4.2: Examples of AR with a tablet on the left and with Google Glass headset on the right

## 4.4 Unity

Unity is a cross platform game engine supporting several application programming interfaces (API) specifically for game development. The key feature of Unity is the ability to deliver for all supported platforms over one project. This dramatically reduces the effort for cross-platform development. Unity has a wide range of SDK support, expanding its capabilities and versatility, including ones specific to AR applications.

## 4.5 Vuforia

Vuforia is a SDK specifically for AR applications, supporting both native development for Android and IOS platforms and also cross-platform development through Unity game engine. Vuforia eliminates the need of any complicated infrastructure by providing embedded algorithms within the SDK handling image recognition and tracking along with all other required tasks for rendering desired information on top of the real-life environment.

Vuforia offers both image and object recognition. Any image can be used as a target as long as the image has sufficient features to be recognized. Features are extracted by online development system of Vuforia and desired image is ranked accordingly to indicate whether it is suitable to be a target for tracking. Object recognition is essentially for 3D tangible objects. Any specific object with adequate features such as being opaque, rigid and one solid piece or having as minimum moving parts as possible can be defined as a target. In order to register a tangible object as a target, Vuforia offers application and a guideline for scanning rather than requiring a 3D model data of the object. This feature makes the Vuforia more accessible for various application scenarios.

Detailed information on utilization and functionalities of Vuforia is provided in Chapter 6.

## 4.6 Data Types

Data types might sometimes be confusing and various interpretations are made. This section is intended to clarify any probable confusion related to data types throughout the thesis work.

At the highest level, there are two types of data categorized as quantitative and qualitative.

### 4.6.1 Quantitative

Quantitative data is associated with objectively measured numbers. This type of data falls into two subgroups as continuous and discrete.

- Discrete data is the quantities that can be counted rather than be measured. Those quantities can not yield any more precision than the quantitative number they are assigned.

- Continuous data, unlike discrete data, can be measured and the precision of the obtained number is likely to be further improved.

### 4.6.2 Qualitative

Qualitative data is the classification or categorization of the data rather than measurement. Qualitative data has three different subgroups as binary, nominal and ordinal.

- Binary data provides two counter states such as yes/no, true/false.

- Nominal data is the categorization of each item by assignment of a name without any specific order or reasoning. A list of country names would be a nominal data.

- Ordinal data type is essentially ordered nominal data. When the order matters, ordinal data type is used.

# Chapter 5

# Design of Physical Models

This chapter discusses the design rules to be considered in order to achieve valid models both in terms of human perception and data embedding.

Each design discipline has its own standards to be considered when being applied. When information visualization on 2D surfaces, such as screen or paper, is concerned, comprehensive studies by various scholars are available in literature. The most prominent and well-known rules, defined by Jaques Bertin, are the visual variables, shown in figure 5.1 [19]. This set of rules depicts the use of various retinal attributes in accordance with the type of the data of interest; Quantitative or qualitative.

Although those set of rules are defined for 2D visualizations, they can be adopted by 3D tangibles to a certain extend. With the physicalization, introduction of touch sensation other than retinal input requires further study on physical attributes.

In the literature, there are various studies conducted in order to examine the cognitive effects of physical attributes on attitudes of people and perception of information embedded into tangibles. Based on those findings, a specific design methodology is applied to each model. In the following subsections those methodologies are discussed.

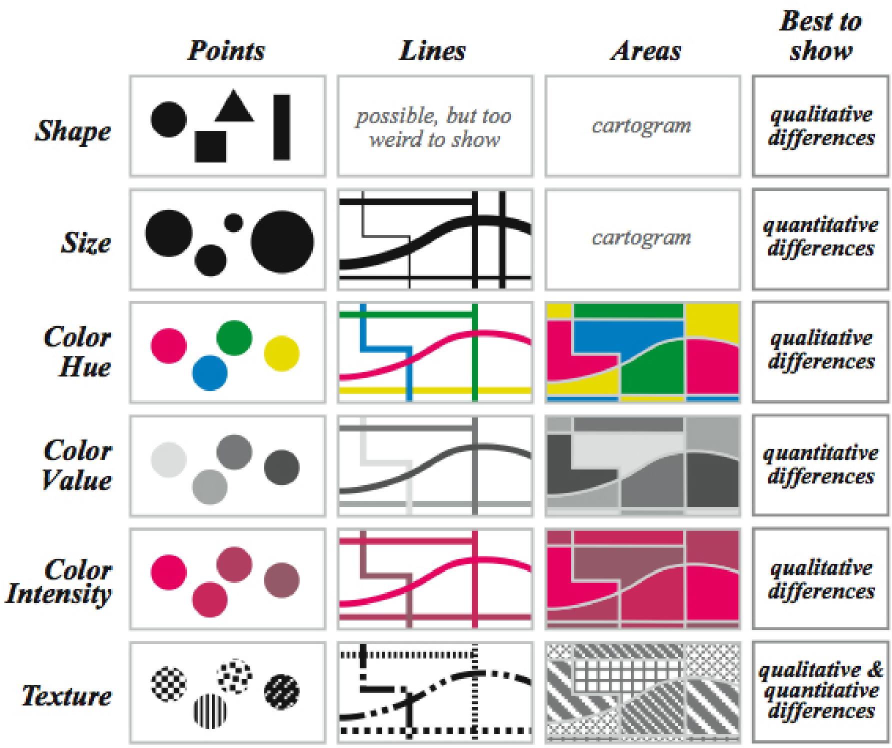


Figure 5.1: Visaul variables defined by Bertin

## 5.1 Model 1: Data Tower

The motivation for this physical model is offering the capability of manifesting both quantitative and qualitative data with an easy to engage design by virtue of its resemblance to a well-known on-screen visualization method. Data tower is designed to be a model for embracing ordinal type of qualitative data in one axis and both discrete or continuous data in another axis. Besides, with each instance of this model one can also represent a nominal type of data. On the whole, data tower is modelled to be used with 3-Dimensional data and designed for users to feel familiar with.

In that sense, design of stacked 3D geometries on top of each other is proposed. As each stacked instance of a geometry represents a qualitative data, dimensions of each geometry can yield quantitative data. The 3D geometric shape to be used in this model is determined to be a cylinder due to its curved nature. Psychological studies reveal that humans have a tendency to rather objects with curvature than ones featuring sharp edges. Bar and Neta hypothesized that sharp outlines of an object are perceived as an intimidating remark. The results of the conducted study verified the accuracy of the theory that humans are verily inclined to show positive reactions toward objects featuring curved properties [47].

Figure 5.2: A set of primitive geometries featuring curved and sharp contours

A non-elliptical cylinder has two parameters to be defined in order to form a 3D geometry; Radius and height. A quantitative data value can be assigned either of those parameters. A set of stacked cylinders with a constant radius and varying heights would require differentiation of each cylinder by another feature such as color or texture, which introduces useless and undesired complexity. However, use of radius for representation of quantitative data with a fixed height parameter does not require such differentiation of each cylinder and allows effortless perception. In figure 5.3 an example of stacked cylinders with fixed height of **h** and varying radii of **R1**, **R2** and **R3** is shown.

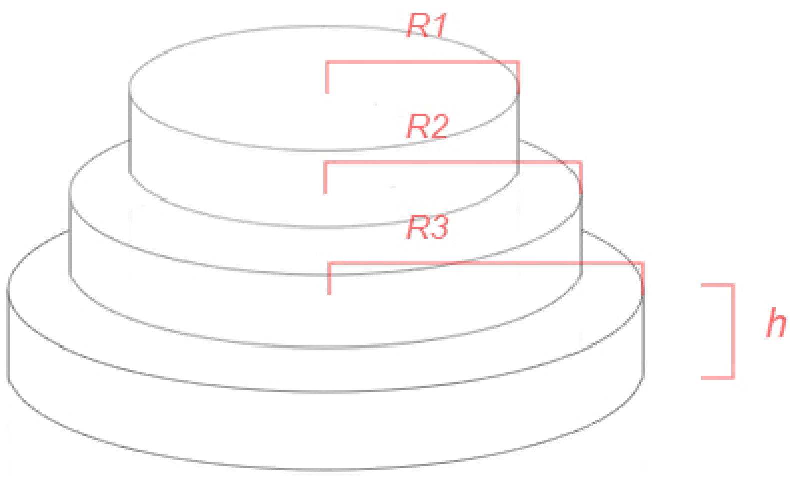


Figure 5.3: Illustration of stacked cylinders with varying and fixed parameters

As quantitative data is represented by the radius of each cylinder, each stacked cylinder instance represents an ordinal qualitative data. Although a nominal data can also be represented by each cylinder, it would require proper labelling for a practical use. Therefore, representing nominal data with multiple samples of data tower is considered to be more appropriate. The key feature of data tower model is the ease for users to straightforwardly relate it to the well-known bar chart model. In figure 5.4 resemblance of each design is shown.

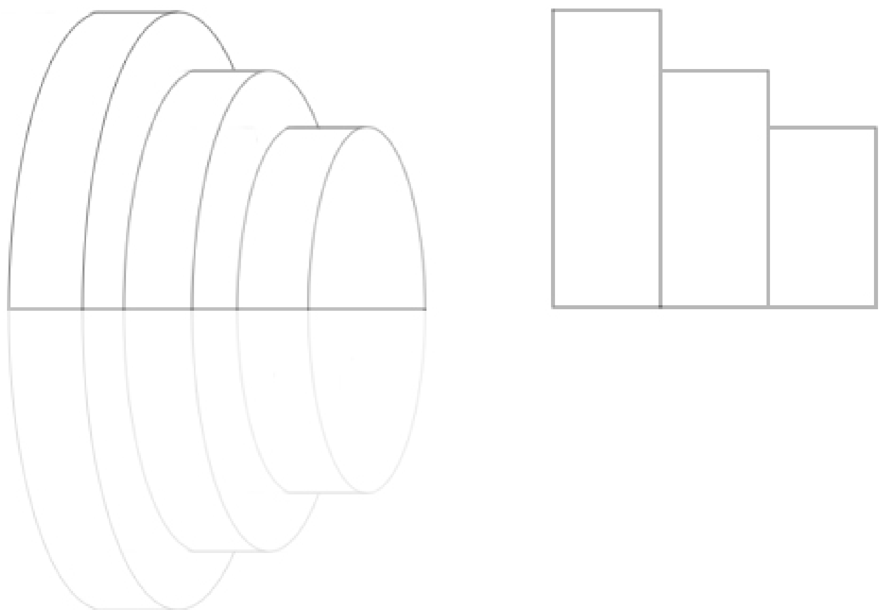

Figure 5.4: Resemblance of Data Tower and Bar Chart

When suitable, multiple number of data sets can be compared and contrasted. With the idea of fusing several data sets in one data tower model, a full cylinder is allowed to be divided into maximum of four slices, consequently holding four different data sets. The most prominent use case of this option is imitating 2D four-fold graph. Two different examples of four-fold graph are shown in figure 5.5. Graph on the left hand side shows the admissions and rejections to graduate school at Berkeley in accordance with gender in 1973 [48]. The one on the right hand side depicts the weight lost depending on four possible combinations of two different types of exercises with being on diet or not [49].

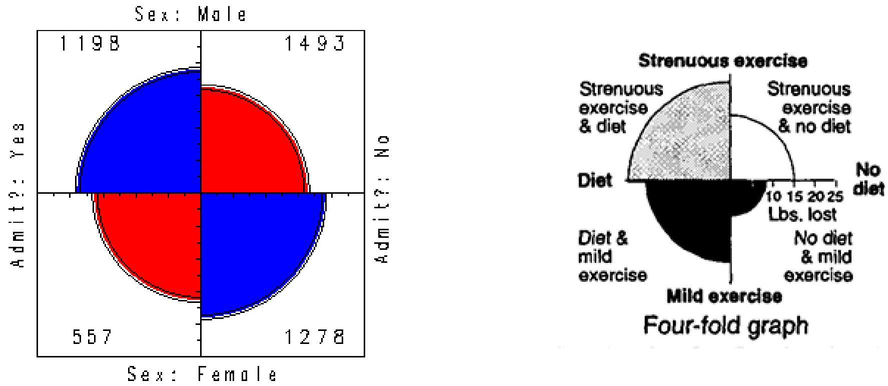


Figure 5.5: Two different use cases of four-fold graph

Design of data tower allows division into two, three and four segments as shown by a 3D render in figure 5.6.

Figure 5.6: One data set along with fusion of two, three and four data sets respectively

The data tower model is designed to be 3D manufactured. Given the fabrication capabilities of standard 3D printers, this model is highly suitable to be precisely fabricated with its low geometry complexity.

To summarize, basic properties of the model are listed as follows;

- Use with 2D or 3D data
- Fusing up to 4 data sets
- Resemblance to Bar chart model
- Suitable for 3D printing

## 5.2 Model 2: Data Circles

Data circles are meant to utilize laser cutting or CNC milling machines and designed accordingly. As in the previous model, use of curved outlines are also taken into account for this second model design. Therefore, circle is selected to be the base geometry as 2D conjugate of a cylinder. Unlike data tower, this second model is designed to be unprecedented and therefore shows no resemblance to any available visualization models. As illustrated by Isola et al. [30] and Borkin et al. [29] in Chapter 2, unique visualizations are likely to be more memorable.

With Data Circles model, three different dimensions are available to store information. One dimension is capable of showing either both discrete or continuous type of quantitative data or binary type of qualitative data. Both of other two dimensions are able to store nominal or ordinal type of qualitative data.

The basic principle of the Data Circles model is spreading up to six circles, representing quantitative values of nominal or ordinal qualitative data, around a base circle. Those circles are called 'attribute circles'. Each instance of the model represents nominal or ordinal data as well. Instances are designed to be stackable and in case of use of ordinal data, a trend can be displayed for each attribute. In order to see through attribute circles, they are designed as tube geometries with a wall thickness. Since stacked models are not adhered to each other for forming a permanent solid, two parallel holes are deployed to be used with a special 3D geometry holding all instances without allowing any rotation. Furthermore, a distinctive mark is also added to ease the matching of attribute circles accurately when stacked. Data Circles model also deploys labelling for both attributes and each instance of the model. Figure 5.7 shows the aforementioned details.

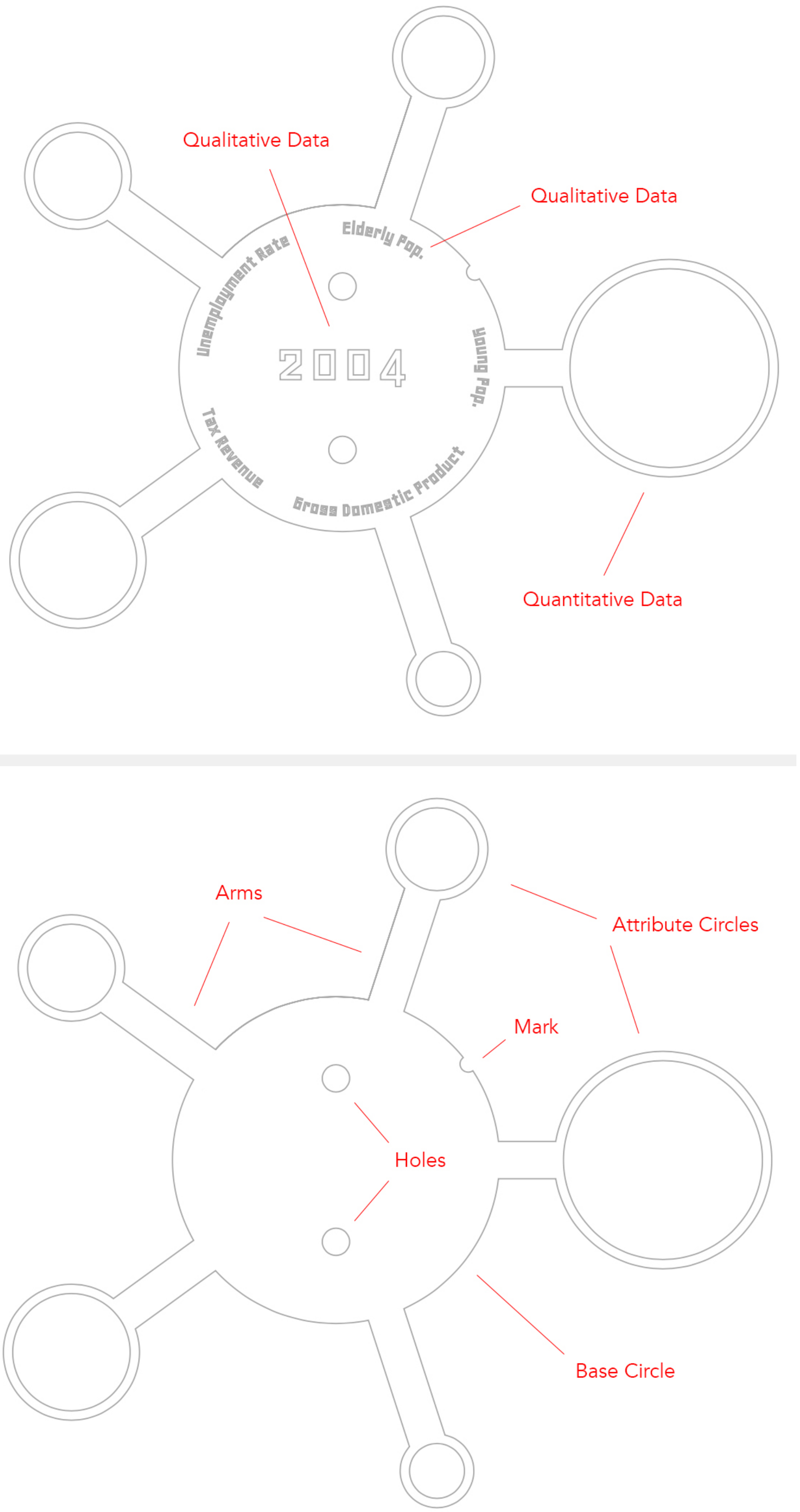


Figure 5.7: Illustration of data circles model showing its features

The radius of the base circle controls how far the attribute circles are positioned from the base circle. Attribute circles are uniformly distributed around the base circle, connected with arms. The width of the arms determines the lower limit for the quantitative data of interest. The radius of the base circle, again, determines the upper limit. The detailed information on use of those limits is explained further in Chapter 6.

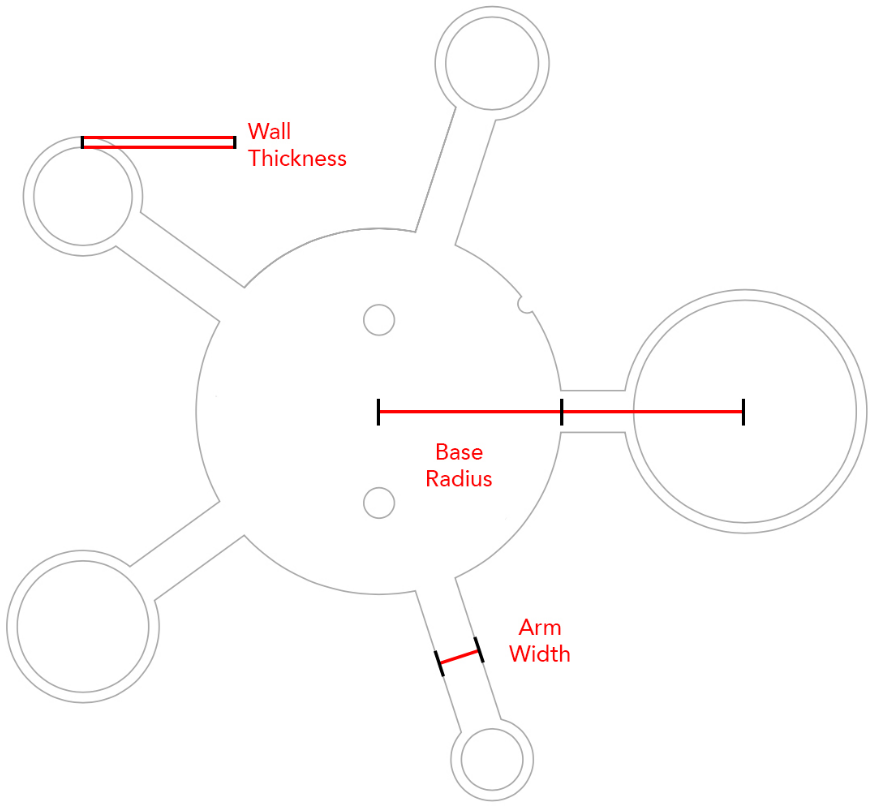


Figure 5.8: Basic dimensions of the model

When use of circles concerned, prominence of area as the varying parameter should not be neglected [50]. Varying the radius in accordance with the value to be visualized results in quadratic growth of the perceived size of the circle, which results in misleading sizing. In order to avoid this misinterpretation, values to be visualized should be considered as the area of circles and unique radii should be derived. This is explained thoroughly with formulizations in Chapter 6. An example displaying the phenomenon is shown in figure 5.9.

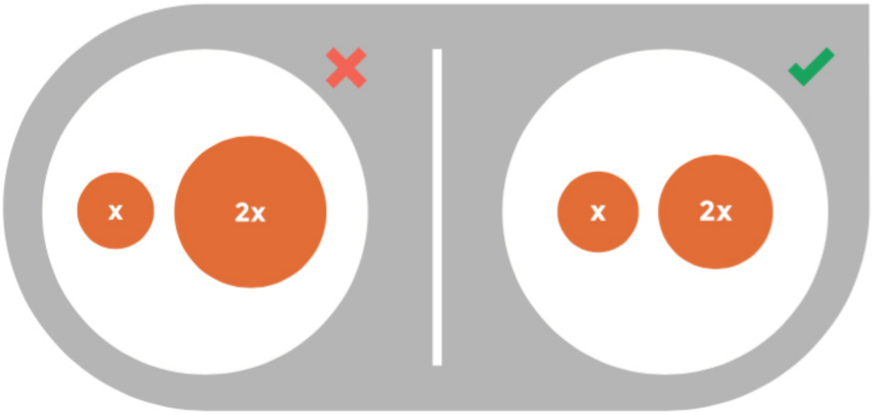


Figure 5.9: Illustration for correct use of circles in visualization [5]

It should be noted that for Data Tower model, use of area is not appropriate due to the fact that stacked cylinder geometries are not meant to be compared according to their surface area size from top view, but rather to be compared in accordance with their radii from side view as in the case of bar chart model.

In summary, properties of Data Circles model as follows;

- Ability to visualize 2D or 3D data
- Stackable design
- Unique design for encouraging memorability
- Suitable for laser cutting or CNC milling

## 5.3 Cognitive Aspects of the Designs

Physical visualizations are unique in terms of the interaction they offer compared to on-screen equivalences or alternatives. The impact on human perception is the key for utilizing their potential. This merely depends on the way the data is depicted through the design of the physical model. Effects of physical objects on human perception is a well-studied research topic and literature features various aspects to be considered when designing physical models.

As illustrated in Chapter 2, memorability is a crucial quality for visualizations. Borkin et al. [29] finds instinctive outcomes that human recognizable objects contribute to memorability of a visualization. On top of that, it is a scientifically proven fact with psychology studies that plain and uncomplicated visualizations support comprehension [31, 32]. Moere et al. supports these findings by showing that styling in information visualization has no significant effect on human cognition [51]. However, another study reveals the fact that visualizations with extra decorative elements perform better than unembellished visualizations in terms of remembrance [52]. This contradiction prompts the encouragement of adopting distinct design approaches on two different models.

In that regard Data Tower model is designed as simple as possible without any decorative element such as featuring complex geometries, multiple colors for each geometry varying in size or text elements as labels for revealing any detail of the data of interest. Moreover, as illustrated in Section 5.1, use of soft edged geometry and resemblance to one of the well-known on-screen visualization types is adopted in order to support other cognitive findings observed by M. Bar et al. and Borkin et al. respectively [29, 47].

Data Circles model, however, offers a more distinctive and unusual design by combining basic geometries, again featuring smooth edges, in a systematic way based on defined design rules in section 5.2. Use of text elements, in order to label the attribute names, appears as decorative elements with the intention of improving the memorability. The study conducted by Gwilt et al. [28] shows that explaining the function of physical models increases the prominence of the physical visualization. Considering the unprecedented design of Data Circles model, it might be required to provide further explanations on the functions of the model in order for users to utilize its full potential. The stackable nature of this model generates a third dimension for observing trends of each attributes and hints at a similar form to Data Tower model.

The physicality in both models takes the advantage of haptic perception. It is studied that haptic perception is more effortless compared to vision in terms of comprehension [53]. Moreover, manipulability of physical objects as a natural interaction enhances the process of learning [54–56].

Considering the results of the aforementioned studies, introduced models have both similarities and distinctions. This is due to the fact that human perception is highly subjective and contradictive results are likely to emerge as illustrated in the beginning of this section. By having diverse design features in both models it is targeted to adopt findings of different cognitive studies.

# Chapter 6

# Authoring Tool and AR Interface Implementation

This chapter thoroughly covers the implementation of the proposed authoring tool and augmented reality interface. Section 6.1 explains the detailed implementation of the authoring tool named 'PhysVis' and elaborates on functionalities. On top of that, section 6.2 covers the design of the augmented reality interface and implementation of the mobile application.

## 6.1 Authoring Tool: PhysVis

Physical visualizations are appealing yet require laborious processes to be designed and fabricated. This is presumably one factor confining both the number of research focusing on physical visualization subject and the use of physical visualizations in various settings including daily life. There are three major obstacles through the process of creating a physical visualization from a data set. First one is modifying the data so that it is suitable to work with a physical model. The second and the most challenging obstacle both in terms of time and work required is the process of generating a physical model in a computer environment. Last problem is the fabrication of the generated model. However, this is lately becoming less of a problem with the growth of rapid prototyping technologies.

PhysVis is a full pipeline converting a data input into an output file ready to be rapid prototyped. To be able to use the PhysVis, only requirement is configuring the data in accordance with a given specific formatting and store it as a CSV file.

Figure 6.1 illustrates the overall software flow which starts with a CSV file input. Then the input data is parsed to generate and display the user specified model with default parameters. Next step offers user interaction with further adjustment of each parameter. Once the desired visualization is obtained, an output file, in accordance with the selected model, is generated for fabrication.

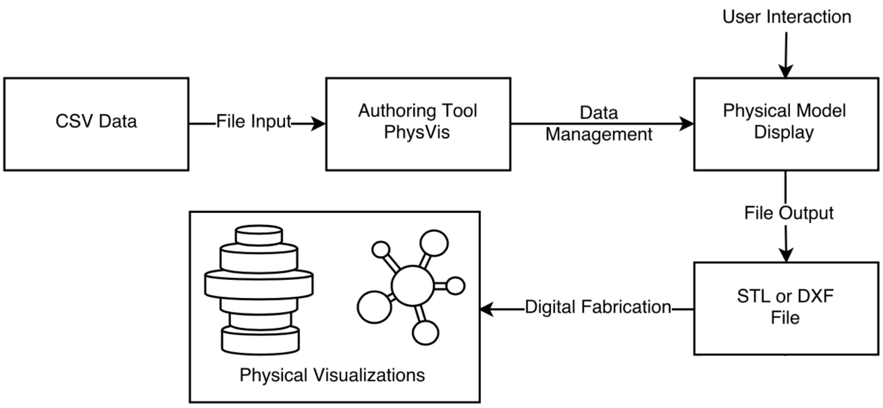


Figure 6.1: A block diagram illustrates the software flow

### 6.1.1 Enabling Libraries

PhysVis is a client-side browser-based cross-platform software implemented in plain JavaScript. A variety of open-source libraries are utilized to introduce various functionalities. In the following subsections, use of each principal library is described.

#### PapaParse

PapaParse is a handy in-browser CSV parsing library. The table form of the CSV file is directly parsed into a two dimensional array for easy access and further manipulation of the data of interest. PapaParse offers some key features such as auto delimiter detection and most importantly dynamic typing. The latter feature distinguishes type of parsed data and stores accordingly. This is useful in terms of elimination of extra type casting. In the software implemented, PapaParse is utilized for effective parsing of CSV file inputs.

### MakerJS

MakerJS is an extensive library developed as Microsoft garage project for drawing scalable vector graphics (SVG) elements with myriad of functionalities. MakerJS offers both primitive shapes and paths for drawing an SVG geometry. The most remarkable feature of the library is the ability to convert SVG elements into DXF, which is the suitable file format for laser cutting or CNC milling. Since SVG rendering is supported by almost all modern web browsers, make use of MakerJS facilitates both displaying the drawn geometries on software interface and exporting valid output files for manufacturing. MakerJS is utilized for parametrically drawing the Data Signs model in SVG format and then converting to DXF format for outputting.

### ThreeJS

ThreeJS is a convenient abstracted WebGL library for generating and displaying 3D graphics in a web-browser environment. WebGL is an API for creating 3D content in a web-browser and its complexity is abstracts away from the user through ThreeJS offering easy functionalities. It uses both canvas and WebGL for displaying the 3D content. Full advantage of ThreeJS library is taken for generating and rendering 3D graphics of Data Tower model. TrackballControls, a versatile example offered by ThreeJS library, is used for introducing 6-Degree of Freedom interaction within the scene displaying the rendered 3D geometry.

### STLExporter

This is a mighty JavaScript library [57] for converting 3D scene meshes generated by ThreeJS into STL file format, which is suitable for 3D printing. The library converts the ThreeJS scene variable into an STL string variable, which is meant to be stored in a blob object as a representation of the actual file.

### FileSaver

In order to save generated blob objects as an actual file on client-side FileSaver library [58] is utilized. This library is effectively used for saving both STL and DXF files generated by related libraries.

### JSZip

In case of file outputting for both models, several files have to be downloaded at once. This requires packaging all files as one, which is essentially zipping. For zipping files on a browser environment, JSZip library offers a useful API. In essence, the library creates another blob object out of each blob to be included in the zip file. The library is used for zipping each instance of Data Circles model along with its complementary objects.

### DatGUI

DatGUI is a widely used library for creating quick graphical user interface designs with adjustable parameters. Both Data Tower and Data Signs models have several parameters to be user specified. DatGUI seamlessly integrates with those parameters of each design and offers real-time manipulation.

## 6.1.2 Implementation

In this section in depth review of PhysVis's implementation is presented.

Before giving detailed information on parsing and manipulation of CSV data, an explanatory subsection exhibits the specific formatting of CSV file to be used with the tool.

### CSV File Formatting

PhysVis accepts a CSV file as data input. The data should be in the given specific formatting in order to be able to used with PhysVis. The instructions on specifications of the formatting as follows;

An example CSV file is shown below.

| **auto** | **Data1** | **Data2** | **Data3** | **Data4** |
|---|---|---|---|---|
| **2000** | 10 | 20 | 30 | 40 |
| **2001** | 50 | 60 | 70 | 80 |
| **2002** | 25 | 35 | 14 | 50 |

Table 6.1: CSV input file format

First cell of the first row contains the identifier and other cells correspond to names of each attribute for Data Sign model, and for Data Tower model they correspond to each instance.

| auto | Data1 | Data2 | Data3 | Data4 |
|---|---|---|---|---|

Table 6.2: First row of the CSV file

First column contains the names of each instance for Data Signs model. As for Data Tower model, those names correspond to each of the stacked cylinders.

| |
|---|
| **2000** |
| **2001** |
| **2002** |

Table 6.3: First column of the CSV file

Identifier helps specifying the lower and upper limits of data sets. Identifier can only be **'auto'**, **'manual'** or **'%'**.

In order for limits to be automatically determined, **'auto'** option is used. In order to specify the limits manually, **'manual'** option can be used. However, that requires additional rows in the CSV data as shown in table 6.4.

| manual | Data1 | Data2 | Data3 | Data4 |
|---|---|---|---|---|
| **2000** | 10 | 20 | 30 | 40 |
| **2001** | 50 | 60 | 70 | 80 |
| **2002** | 25 | 35 | 14 | 50 |
| **Lower** | 10 | 20 | 14 | 40 |
| **Upper** | 50 | 60 | 70 | 80 |

Table 6.4: CSV formatting with 'manual' identifier

Using **'%'** indicates that the given data sets contain percentages and therefore lower and upper limits are automatically set to '0' and '100' respectively.

| '%' | **Data1** | **Data2** | **Data3** | **Data4** |
|---|---|---|---|---|
| **2000** | 10 | 20 | 30 | 40 |
| **2001** | 50 | 60 | 70 | 80 |
| **2002** | 25 | 35 | 14 | 50 |

Table 6.5: CSV formatting with '%' identifier

### Data Management

As illustrated in the previous section, an identifier decides upper and lower limits of each data set. For both of the physical models, there are also user specified upper and lower limits. This is required for linearly mapping the data to the new range, which is specified by physical model parameters. Equation 6.1 shows the relation between each point value of new and raw data, denoted by $D\left[i\right]_{out}$ and $D\left[i\right]_{in}$ respectively. $In_{min}$ and $In_{max}$ denote the lower and upper limits for raw data respectively. Likewise, $Out_{min}$ and $Out_{max}$ denote the target limits.

$$D\left[i\right]_{out} = \frac{\left(D\left[i\right]_{in} - In_{min}\right) \times \left(Out_{max} - Out_{min}\right)}{In_{max} - In_{min}} + Out_{min} \tag{6.1}$$

Upon mapping, new data points are generated to be used as parameters to construct the physical models. A comparison of raw and mapped data points is plotted in figure 6.2. As illustrated, -y axis scale of both plots are different due to use of different ranges, whereas the trend is identical.

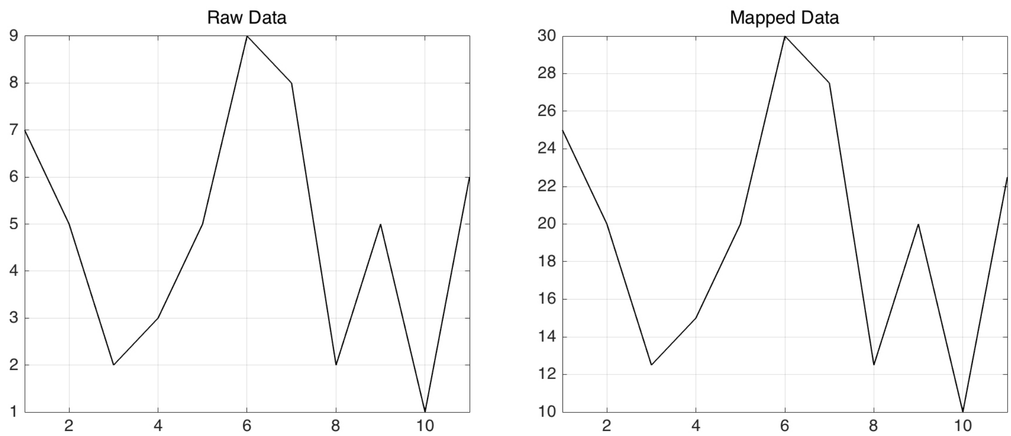


Figure 6.2: Comparison of raw and mapped data

Interpretation of trend for any set of data is import in terms of reflecting a correct sense of perception. A set of data is highly prone to be misleading when not visualized properly. When each point is linearly interpolated, trend is essentially the slope of each line showing a descent or ascent. Since step size of -x axis is fixed, the only parameter altering the slope is the numeric value of two data points, which are linearly interpolated. For a whole data set, scaling each number accordingly could yield the same trend with less emphasis on abrupt rise and falls, hence the elimination of misinterpretation. The factor of scaling should be manually determined by a pertinent user intervention.

A new formulation is suggested for scaled mapping of a set of data. The magnitude of the scaling is selected to be depending on range of the data set. Equation 6.2 shows the new mapping function, where range is difference of maximum and minimum values in the data set and denoted by $R$.

$\alpha$ is the value of scaling factor and $\delta$ is the magnitude of the scaling factor affecting the mapping function. Figure 6.3 depicts the resulting effects of different levels of scaling factors.

$$\begin{aligned}
&R = D_{max} - D_{min} \\
&\delta = R * \alpha \\
&D\left[i\right]_{out} = \frac{((D\left[i\right]_{in} + \delta) - In_{min}) \times (Out_{max} - Out_{min})}{(In_{max} + 2 \times \delta) - In_{min}} + Out_{min}
\end{aligned} \tag{6.2}$$

In essence, equation 6.2 alters the upper limit of data set and shifts up each point of data by $\delta$, thus generates a new wider raw data range, which effectively results in small variances between each data point. Shifting each point up by addition of $\delta$ is necessary in order not to deviate the numbers way too much from the original target range.

As illustrated in figure 6.3, increasing the scaling factor effectively decreases the slope of each line and preserves the trend as desired.

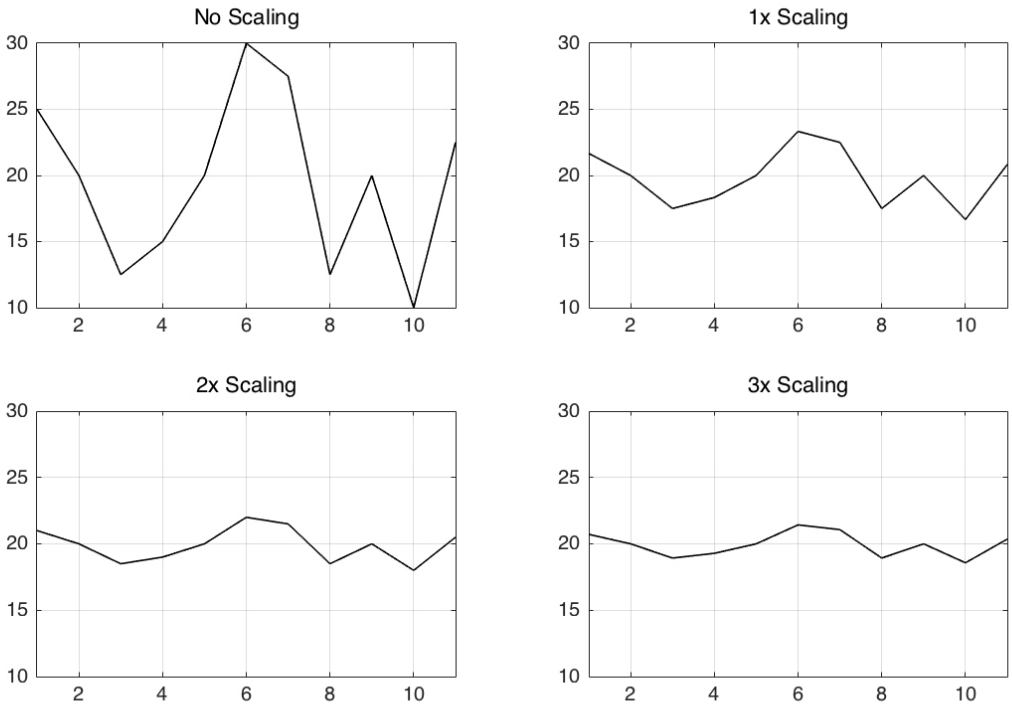


Figure 6.3: Results of different levels of scaling on the trend

### Use of Area

In Chapter 5 it is presented that use of area rather than radius when mapping the data is crucial with regard to accurate visualization. This subsection elaborates on the idea with formulizations.

When visualizing with circular shapes is concerned, make use of area rather than radius conveys a better perception of the numbers compared, meaning that, the numbers to be visualized should not be treated as the radius of the circular region but the area. Therefore, new radius values should be obtained from the given data, which would then be used to draw the circles. This method requires a reference point; a base radius.

Consider the example of two data points given as $[2, 4]$. The ratio between two numbers is 2. If they are considered as the radii of the circles to be drawn, resulting areas for the circular regions would then be, 12.5664 and 50.2655 respectively.

However, if they are regarded as the areas of each circular region, then the new radii would be given as $[r_1, r_2]$, shown in equation 6.3.

$$
\begin{aligned}
r_1 &= \sqrt{\frac{2}{\pi}} = 0.797 \\
r_2 &= \sqrt{\frac{4}{\pi}} = 1.128
\end{aligned}
\tag{6.3}
$$

The ratio between $r_1$ and $r_2$ is 1.415.

If the base radius is selected as the median of the data set, then the new radii would be;

$$
r_i = \frac{\sqrt{\frac{D_i}{\pi}}}{\sqrt{\frac{\widetilde{x}}{\pi}}} = \widetilde{x}\sqrt{\frac{D_i}{\widetilde{x}}}
\tag{6.4}
$$

Where $D_i$ is the data point and $\widetilde{x}$ is the median of the data set. This newly constructed array of radii is then used to draw the circular regions.

For the same example of $[2, 4]$, new areas are calculated as 12.5664 and 25.1323 respectively, which reflects the exact ratio between the data points, which was calculated as 2.

### Generating Models

Each model has its well-defined rules presented in Chapter 5. Those rules are followed when generating the virtual models, in accordance with the managed data sets, to be displayed and outputted eventually.

- Data Circles

Data Circles model is generated as a 2D SVG drawing with primitive geometries of arcs, circles and lines. SVG is the most appropriate type for displaying visuals with any loss of detail by virtue of its scalable essence. Principal trigonometric algorithms are utilized for generating reference points for each single geometry. Then functions of MakerJS library are employed to draw the final parametric shape.

Data Circles model features labeling, which requires use of local font files stored on client-side. Loading a font is an asynchronous computation and therefore a representative variable is required in order not to interrupt the flow of the code regardless of the asynchronous function's state. Promise is one solution offered in JavaScript environment as a representative of the variable of interest. Once a pending promise is fulfilled, handlers queued up by the 'then' method of the promise are called to use the resolved value. 'Then' method is essentially a callback function.

Complete model is displayed in a DIV container defined within the HTML file. In order to generate the downloadable DXF output, another method of MakerJS library is used with defined units. Although displayed model does not reflect the actual dimensions, DXF output is defined in unit of millimeter.

- Data Tower

Data Tower is a 3D mesh generated with ThreeJS library abstracting the WebGL. Since the displayed model is expected to be 3D printed eventually, some extra care must be taken in order for STL output to be compatible with a 3D printing software.

Taking that into account, polygons of the generated 3D meshes are required to have normal vectors facing outwards for proper manufacturing without any errors. Data Tower model is basically stack of cylinders. However, since overlapping surfaces are also undesired, generating it is not as simple. Those constraints impose either use of constructive solid geometry technique with basic geometries and Boolean operations to avoid overlapping surfaces or generating desired geometries algorithmically with polygons. Latter is used to form a 3D mesh.

All the dimensions defining the model are parametric and allowed to be user specified within the defined limits. Each dimension is unitless and interpreted in accordance with the used 3D printing software.

### 6.1.3 User Interface Design

A graphical user interface (GUI) is designed in HTML and CSS to seamlessly work with client-side JavaScript code. Importance of a well-designed GUI is indispensable for supporting a better user interaction experience, which eventually facilitates the use of the tool.

One remarkable technical feature of the interface is uninterruptable switching between pages. All the content is contained in one HTML file so that the user is free to switch between different pages through the sidebar without requiring any page loading. This is achieved by displaying the content of the selected page and hiding the rest with a piece of JavaScript code. The significance of this design becomes clear, for example, when the user wants to check the **Usage** page without terminating the content of the **Data Tower** page. Likewise, user can work on two different visualization methods at the same time without requiring another instance of the page.

Despite the 6 paged content, including the intro page, web-site operates on 3 fundamental pages; **Usage**, **Data Tower** and **Data Circles**.

**Usage** page gives detailed information on CSV file formatting to create suitable databases. A screenshot of the **Usage** page is shown in figure 6.4

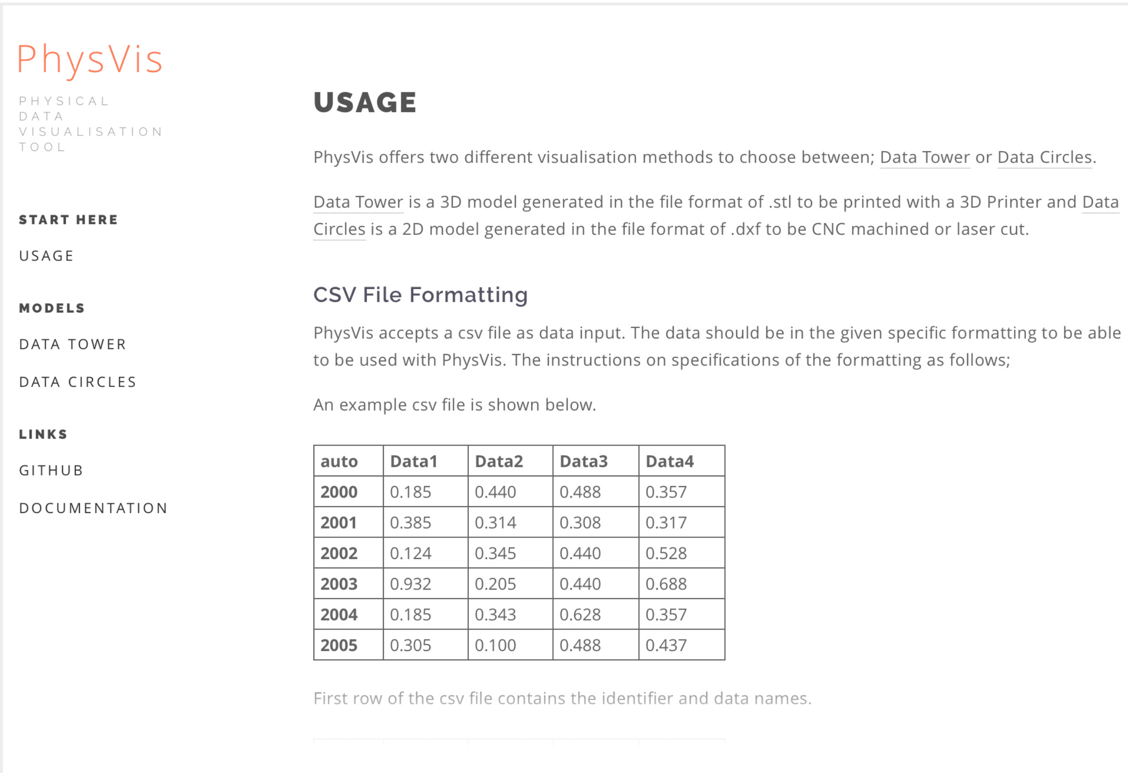


Figure 6.4: Screenshot of 'Usage' Page

Other Two pages give access to data physicalization tools. Screenshots of **Data Tower** and **Data Circles** pages are shown in figure 6.5 and 6.6 respectively. Further details on functionalities of those tools are discussed in the following section.

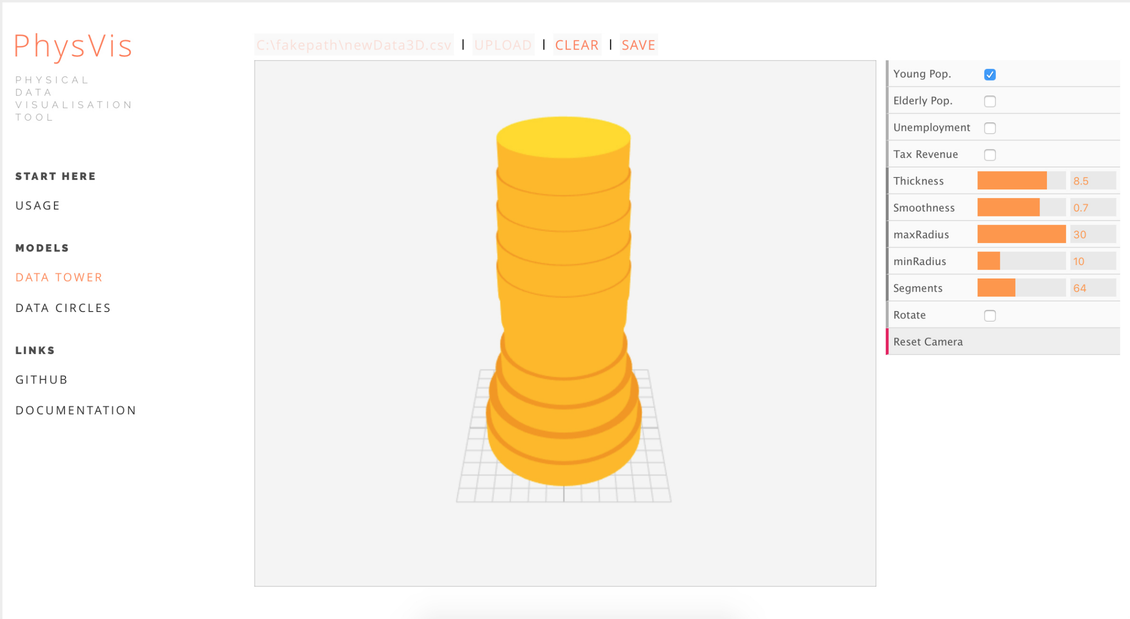


Figure 6.5: Interface for creating Data Tower models

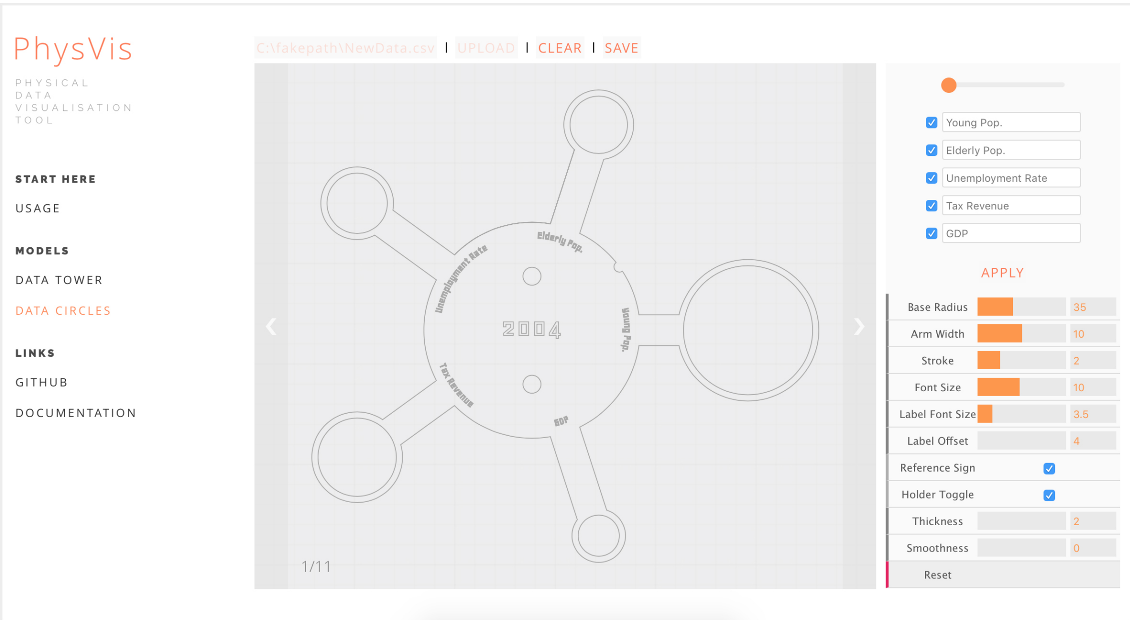


Figure 6.6: Interface for creating Data Circles models

### 6.1.4 Functionalities

PhysVis offers great flexibility in terms of customizability. Almost all the defining parameters of both models are allowed to be user specified within the given ranges in order not to be violated.

#### Aspects of Data Tower

User can define a global height value for all cylinders and determine global upper and lower limits for radius of each cylinder. Those parameters hint the actual size of the tangible object.

A parameter named **smoothness** adjusts the scaling factor denoted by $\alpha$ in equation 6.2.

In figure 6.7 the effect of scaling is shown on the Data Tower model. First and second columns illustrate the resulting line graphs and physical models with no scaling factor ($\alpha = 0$) and scaling factor of 0.7 ($\alpha = 0.7$) respectively.

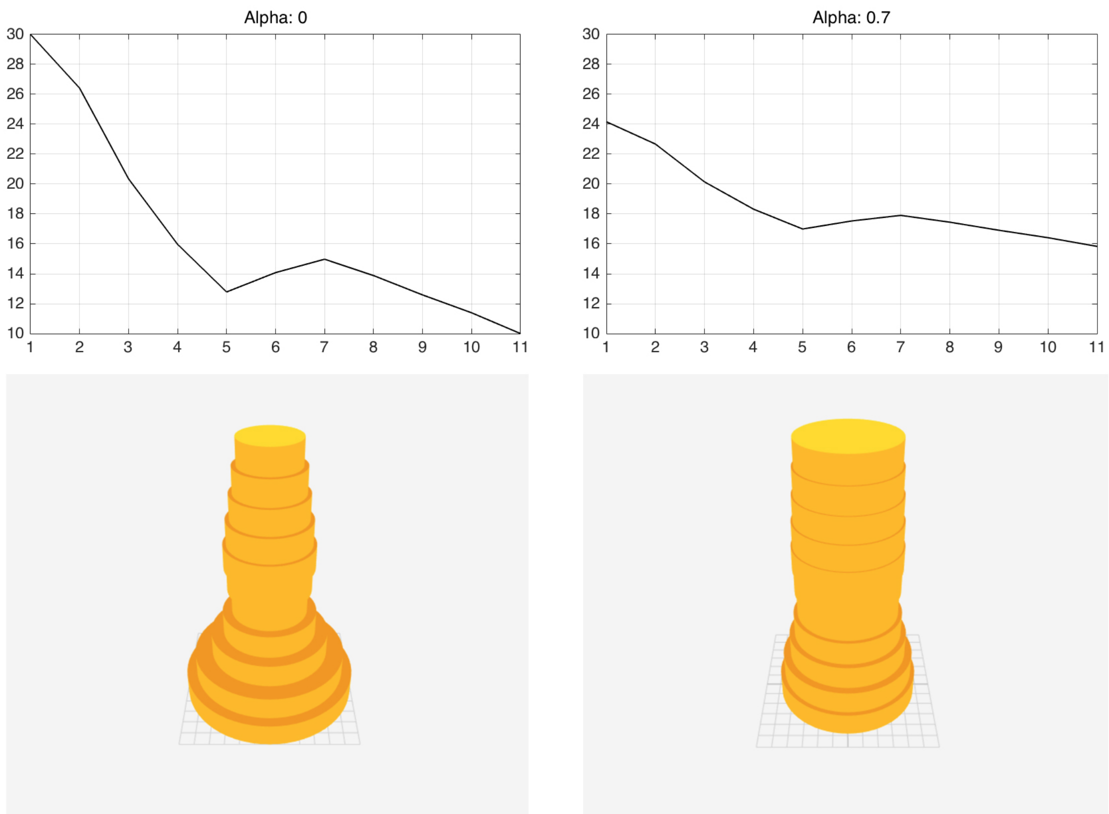


Figure 6.7: Effect of scaling factor on Data Tower model

The fusion of up to 4 data sets appears as an option in the tool and the user can select which models to be fused and the resulting model is displayed in real-time as illustrated in figure 6.8. Each data set is indicated with a different shade of a color.

Another option is the number of segments for generating high quality meshes for the best performance in the fabrication process.


Figure 6.8: Screenshot showing the 4 data fused in one Data Tower model

### Aspects of Data Circles

Similar to Data Tower model, defining parameters of Data Circles model can also be user specified and those parameters directly alter the actual size of the physical output.

The user can decide on the font size of both model name and attribute names. Furthermore, distance of attribute names relative to base circle can also be adjusted.

Users are free to toggle on or off the reference mark and double holes for holding all the instances together with a dedicated 3D model.

The parameter of **thickness** should be accurately specified in order for a 3D holder geometry to be generated. This geometry, shown in figure 6.9, is also generated with ThreeJS library as a 3D printable mesh and included in the ZIP file of Data Circles model output.

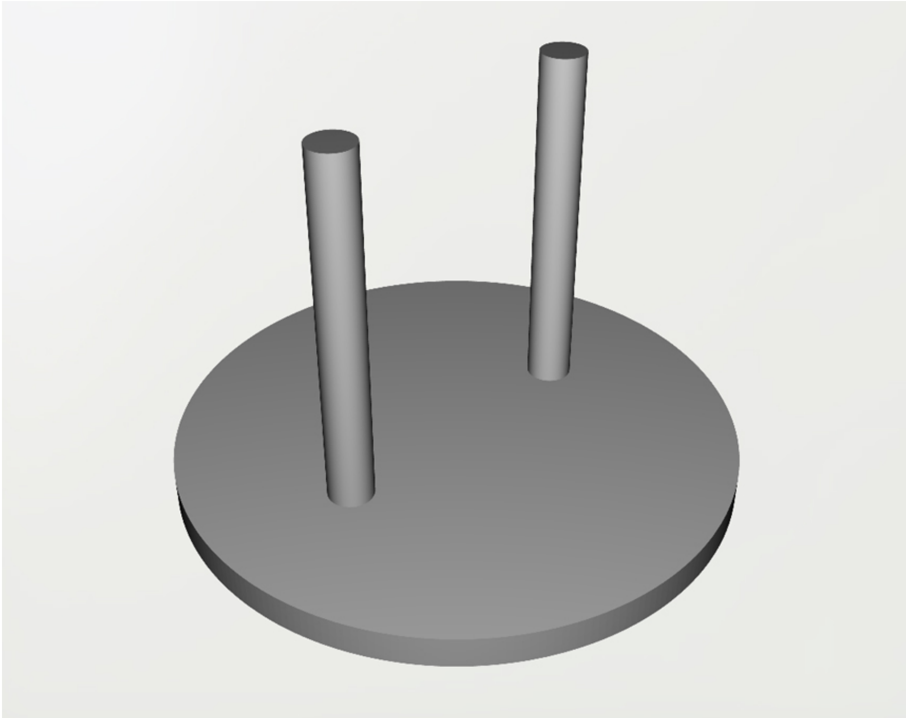

Figure 6.9: 3D holder geometry generated as an STL file to be 3D printed

One facilitating feature of Data Circles model is the real-time attribute name editing and selecting which attributes to be included without need of any CSV file modification.

Last but not least, Data Circles model also has a scaling factor for smoothing the trend of data sets when desired.

### Common Details

Both of the interfaces are designed to be intuitive and user friendly with a flat design featuring minimal stylistic elements and as minimum parametric option as possible in order not to puzzle the user with redundant details.

Each interface prompts a message for user to upload a valid CSV file. Each CSV file upload attempt is followed by an input check process in order to avoid any violations of the CSV file rules. Invalid files are rejected with a relevant error message and followed by a valid file request message.

Once a valid file is uploaded and parameters for the physical model are set as desired, user can download the files with **SAVE** button for manufacturing process. If the user cares for a new CSV file, previous session must be terminated by **CLEAR** button. This action reactivates the **CHOOSE A FILE** button and enables the user to continue with a new CSV file.

## 6.2 Augmented Reality Application

Augmented reality application is developed for further enhancing the capabilities of tangible models beyond physicality with the notion of 'Overview first, detail on demand' adopted by Shneiderman [59]. The

Principally, physicalization is converting digital bits into physical world objects. This can be considered as an open loop system. Introduction of the augmented reality closes the loop by recreating physical objects with additional information in a virtual environment so as to eliminate the limitations of physical object introduced by the materiality. Both open and closed loops systems are illustrated in figure 6.10.

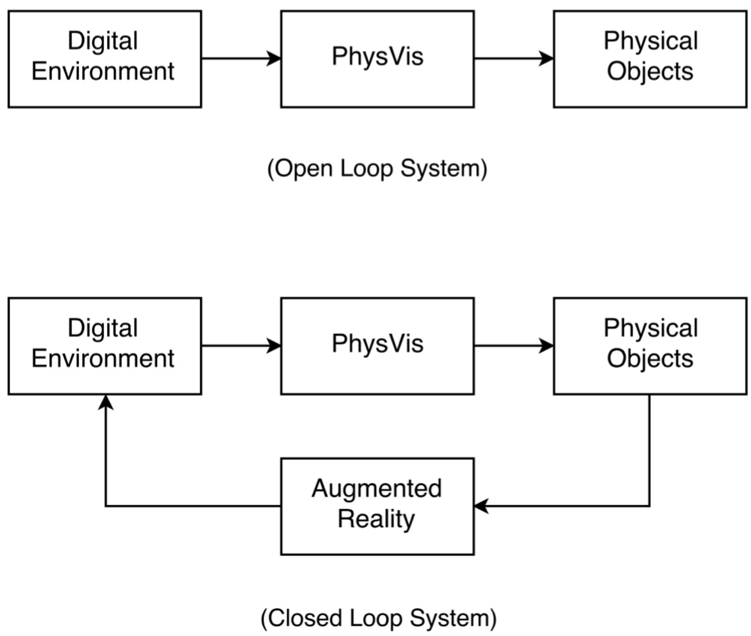


Figure 6.10: Open and closed loop systems with and without augmented reality respectively

### 6.2.1 Interface & Interaction Design

Both interface and interaction of the augmented reality application is a proof of concept design to demonstrate the potential use with physical visualization objects.

Nevertheless, a fully functional, minimalist and intuitive interface is developed with Vuforia SDK on Unity. Both Data Tower and Data Circles models are compatible with augmented reality application and both interactions are intrinsically akin to each other.

### AR Design for Data Tower

Data tower model features stacked cylinders, each representing a quantitative data by its radius. Although the trend is perceivable without need to display any explicit numeric value, detailed information may matter in some cases. Besides, unlike Data Circles model, this model features no labeling. Considering all, AR interface comes in handy by augmenting the physical reality with that hidden piece of information.

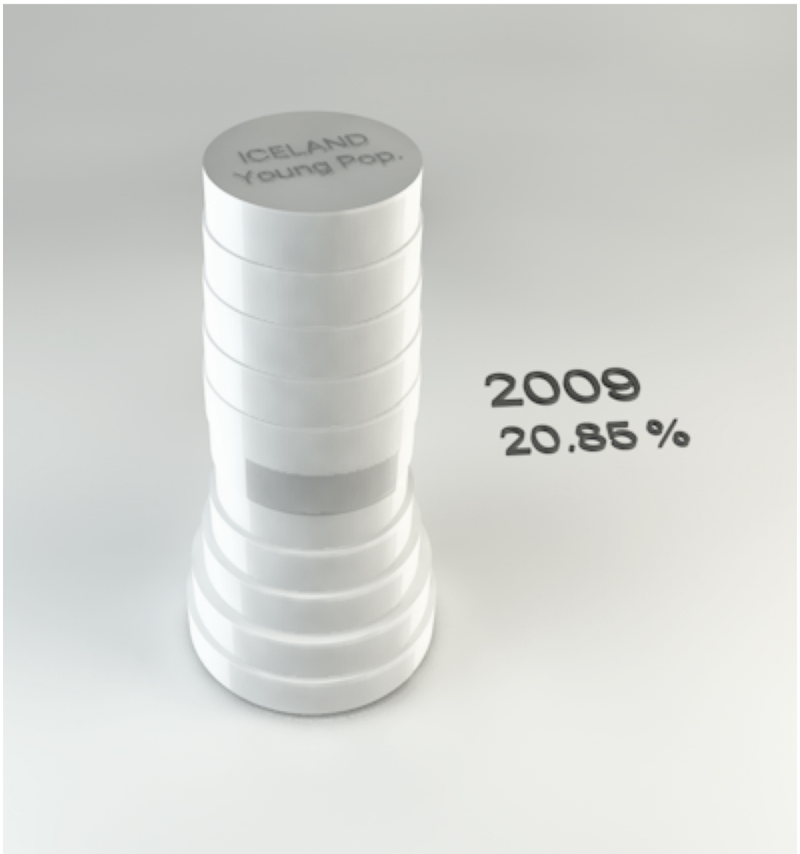


Figure 6.11: A render view showing the AR interface design for Data Tower model

A curved polygon with proper dimensions in accordance with radius of each cylinder is superimposed on each of the cylinders to act like a button to initiate an action. Those buttons are designed with a translucent material in order not to completely hide the details of the physical object as exhibiting their presence. Name of the specific attribute is displayed on top of the physical model.

Figure 6.11 shows a 3D scene render to hint the interface design of the AR. Translucent buttons are shown with light grey material and the selected button becomes darker to indicate its activity. Name of the selected data cylinder is displayed next to the highlighted button along with the exact value represented by the individual cylinder.

The interaction with the interface is intuitively easy. User touches the screen and starts dragging the finger to switch between different buttons related to each cylinder. Each button reveals the related information and indicates its activity by getting highlighted. Releasing the finger does not trigger any other action. A touch to any other region of the screen conceals the active information and deactivates the button by changing its color to default. If the user touches to another button as any of them are active, the one selected become active and the previously selected button becomes inactive along with the texts tied to buttons.

### AR Design for Data Circles

The AR interface design for Data Circles model shows similar traits to previous design with regard to interaction. Unlike Data Tower model, Data Circles model has several instances to be stacked in order to form a 3D geometry akin to Data Tower. In absence of any of the instances, the whole design would be futile. With that motivation, the interface is designed to visualize the rest of the information stored by remaining instances of the Data Circles model. On top of that, explicit values of each attribute circle is also displayed.

Each virtual circle is a replica of the corresponding actual attribute circle on the model with identical dimensions. The model illustrated in figure 6.12 is a demonstration for only one attribute. The interface offers augmentation for each attribute on the model. The interaction with the model is identical to interaction designed for the previous model.

3D render of the suggested interface design is shown in figure 6.12

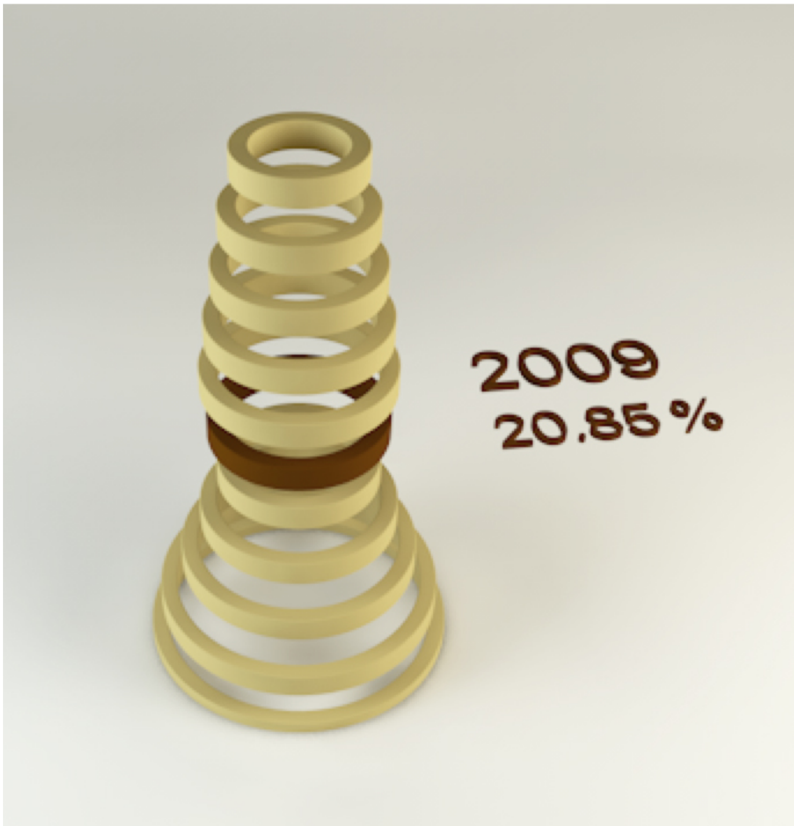


Figure 6.12: A render view showing the AR interface design for Data Circles model

### 6.2.2 Implementation of the Mobile AR Application

The mobile application of augmented reality is implemented with Vuforia SDK on Unity. The project is converted into an Xcode project to be exported. The Xcode project is then compiled and installed on an iPad device for demonstrating the actual functionality.

Since Vuforia handles image recognition and tracking with its built-in algorithms, the only task is modeling 3D geometries and developing scripts in accordance with the design of interface and interaction.

On Unity several scripts are implemented with C# language to deliver interactivity to superimposed 3D geometries. For registering touch and drag gestures on a mobile tablet device, a ray-casting algorithm is implemented.

All the superimposed 3D meshes are generated in a 3D modeling software called

CINEMA4D and then imported into Unity to be used with Vuforia SDK.

Vuforia requires markers with rich features to perform quality tracking. In order to create unique markers, an online augmented reality marker generator tool is used [6]. Figure 6.13 illustrates different augmented reality markers for each model.

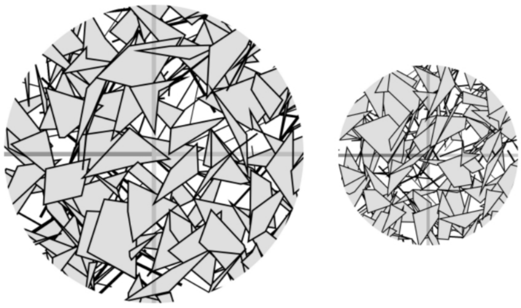

Figure 6.13: Augmented reality markers generated with the dedicated online tool [6]

Since Vuforia offers multiple marker tracking, one mobile application can recognize several markers and display content accordingly.

It should be noted that the augmented reality application is hard-coded and not included in the physical data visualization pipeline. The sole purpose of the mobile application is demonstrating the proof of conceptual use of augmented reality technology in data physicalization subject and give insight for further research.

## Chapter 7

# Results and Discussion

This chapter demonstrates the fabricated physical outputs generated by the implemented pipeline and discusses the potential use of physical visualizations in various settings. In order to evaluate the functionality of the tool and the augmented reality application, a CSV file is generated with some sets of data acquired from the web-page of Organization for Economic Co-operation and Development (OECD). Iceland is selected to be the country of interest and data sets for the following attributes are obtained for each year between 2004 and 2014;

- Young Population

  People under the age of 15 measured as a percentage of total population.

- Elderly Population

  People over the age of 65 measured as a percentage of total population.

- Unemployment Rate

  Number of unemployed people as a percentage of total workers.

- Tax Revenue

  Income of a government through taxation measured as US dollars per capita.

- Gross Domestic Product (GDP)

  Market value of services and goods measured as US dollars per capita.

| auto | Young Pop. | Elderly Pop. | Unemployment Rate | Tax Revenue | GDP |
|---|---|---|---|---|---|
| **2004** | 22.45 | 11.72 | 2.99 | 17165 | 35232 |
| **2005** | 22.09 | 11.73 | 2.55 | 22434 | 37074 |
| **2006** | 21.48 | 11.59 | 2.83 | 22791 | 38653 |
| **2007** | 21.04 | 11.54 | 2.24 | 26652 | 40745 |
| **2008** | 20.72 | 11.50 | 2.94 | 19343 | 42417 |
| **2009** | 20.85 | 11.77 | 7.22 | 12855 | 40967 |
| **2010** | 20.94 | 12.14 | 7.55 | 13899 | 38414 |
| **2011** | 20.83 | 12.46 | 7.03 | 15826 | 39466 |
| **2012** | 20.70 | 12.78 | 5.98 | 15668 | 40486 |
| **2013** | 20.58 | 13.08 | 5.38 | 17184 | 42670 |
| **2014** | 20.44 | 13.38 | 4.89 | 20418 | 44331 |

Table 7.1: CSV file formed with OECD data

The exact CSV file is shown in table 7.1. This CSV file is deployed for generating STL and DXF files, which then be used for digital fabrication of the physical visualization objects.

## 7.1 Requirements

In this section a list of requirements is defined to be satisfied by PhysVis and AR application.

- **R1:** Generate 'Data Tower' model
  - **R1.1:** Demonstrate AR application on 'Data Tower' model
- **R2:** Generate 'Data Circles' model
  - **R2.1:** Demonstrate AR application on 'Data Circles' model

The requirements listed above are defined to reflect the feasibility of PhysVis and AR application in real-life. The following section demonstrates the results as a validation of the given requirements.

## 7.2 Validation

In this section requirements are validated sequentially. Validation of each requirement are made with the CSV file shown in table 7.1.

#### R1: Generate 'Data Tower' model

The dedicated CSV file for demonstration is used with the following parameters from the control panel of the tool, shown in figure 7.1, in order to generate STL files to be used in the process of manufacturing.

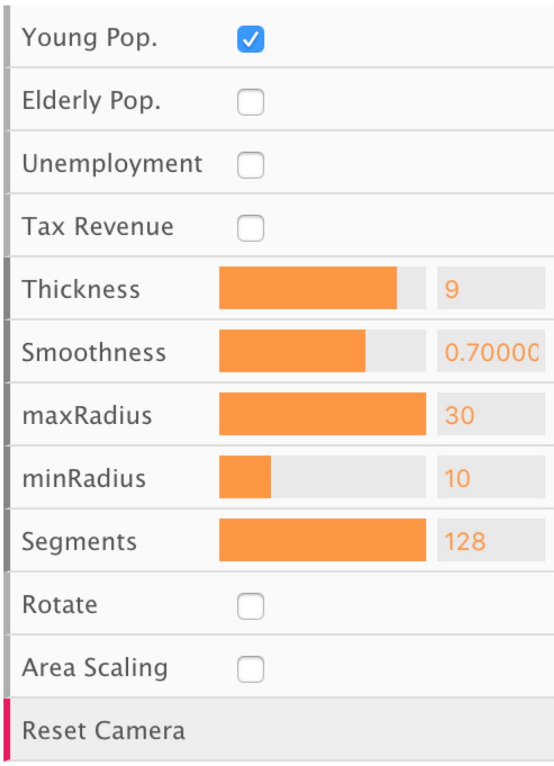


Figure 7.1: Parameters for generating Data Tower model

**Young Population** attribute is selected to be physically visualized with thickness of 9 millimeters for each cylinder. Consequently, the height of the actual physical model is 99 millimeters, since there are 11 data points in the data set. The range of the radius for cylinders are selected to be between 10 and 30 millimeters. Scaling factor of $\alpha$ is selected as 0.7 in order not to exaggerate the trend of the data. Segments option is set to 128 in order to generate a model with highest quality.

Figure 7.2 shows the actual photo of the fabricated physical object along with 3D mesh generated by PhysVis in STL format. The fabricated physical model features the marker image on top of it for purpose of AR application.

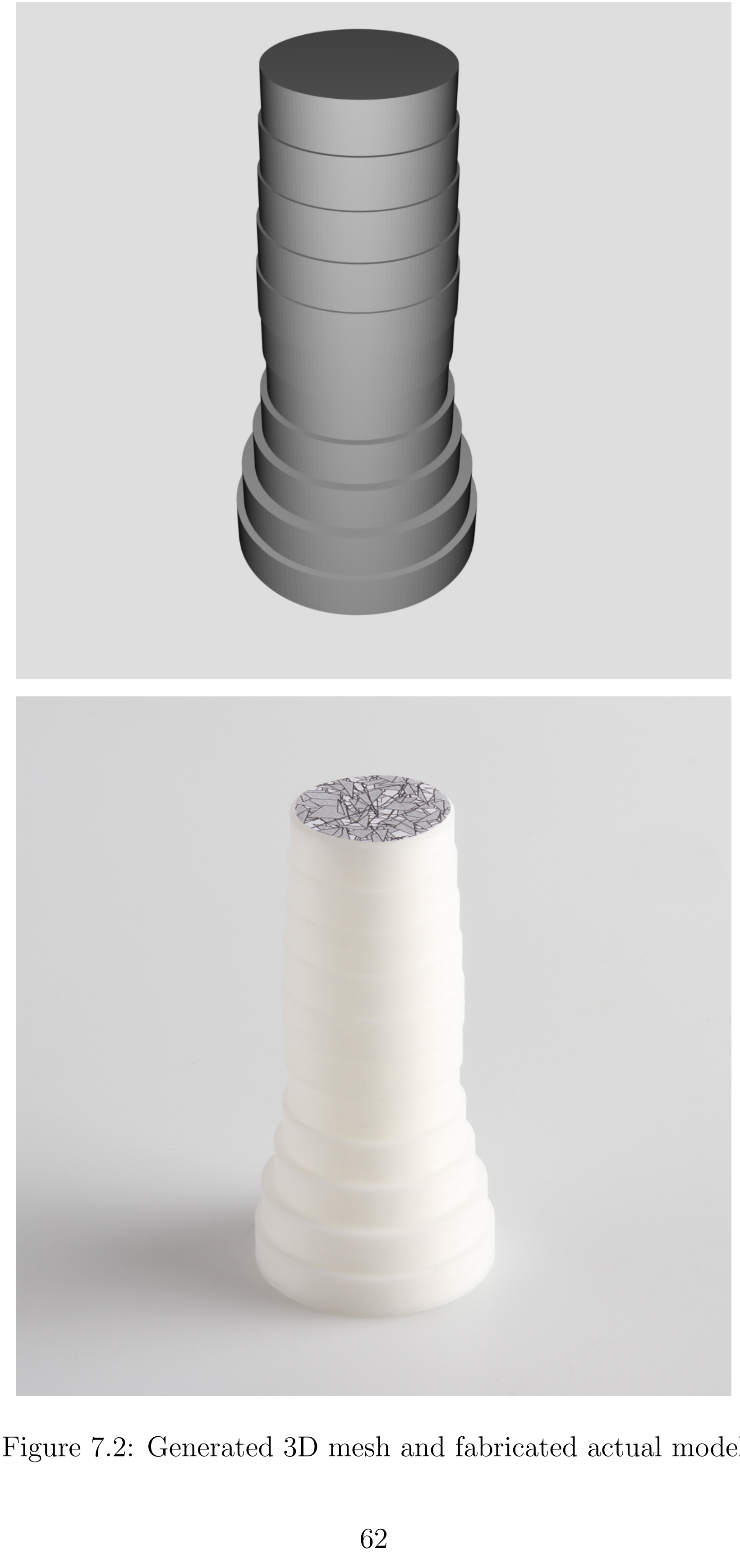

Figure 7.2: Generated 3D mesh and fabricated actual model

The requirement denoted by **R1** is met by generating a 3D mesh file in STL format and then fabricating the model with a 3D printer. Fabrication of the physical object takes approximately 8 hours. This duration depends both on parameters defining the actual size of the object and various parameters of the 3D printing software.

### R1.1: Demonstrate AR application on 'Data Tower' model

The proposed design in section 6.2.1 is implemented in Unity and run on an iPad device. The interface shown in figure 7.3 verifies the implementation of the interface and confirms the practical use on a manufactured physical object generated in PhysVis.

By this demonstration, requirement denoted by **R1.1** is fulfilled.


Figure 7.3: Demonstration of proposed AR interface on Data Tower model

### R2: Generate 'Data Circles' model

The same CSV file is again used for generating several DXF files along with the 3D holder object generated in STL format to be 3D printed. The parameters showed in figure 7.4 are set from the control panel of the tool. All 5 attributes are included in the model. The thickness of the wooden sheets to be used with laser cutter is 4mm, hence the thickness parameter of 4mm.

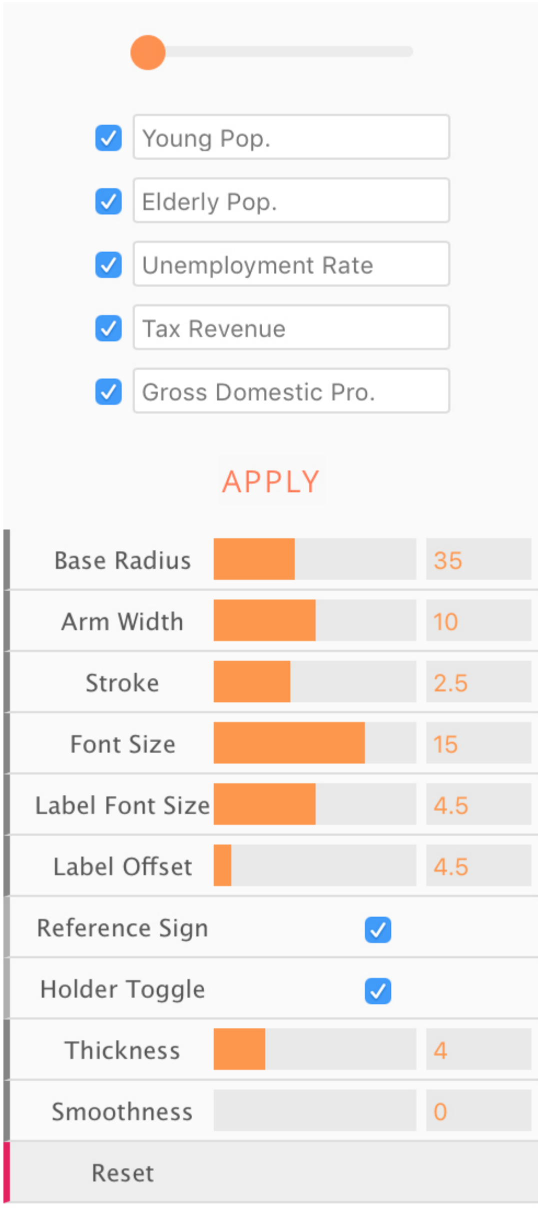


Figure 7.4: Parameters for generating Data Circles model

A screenshot from a DXF viewer software, shown in figure 7.5, displays one of the outputs generated by PhyVis. This DXF output along with other 10 instances are manufactured with a laser cutting machine. For all 11 instances it takes approximately 2 hours to complete the fabrication process. The time elapsed during the manufacturing depends on several variables such as speed of the laser cutting machine, dimensions of the generated DXF files and details on the model. It is observed that the process of engraving is more time consuming than cutting. The 3D mesh of the model holder object generated by PhysVis is 3D printed to be realised.

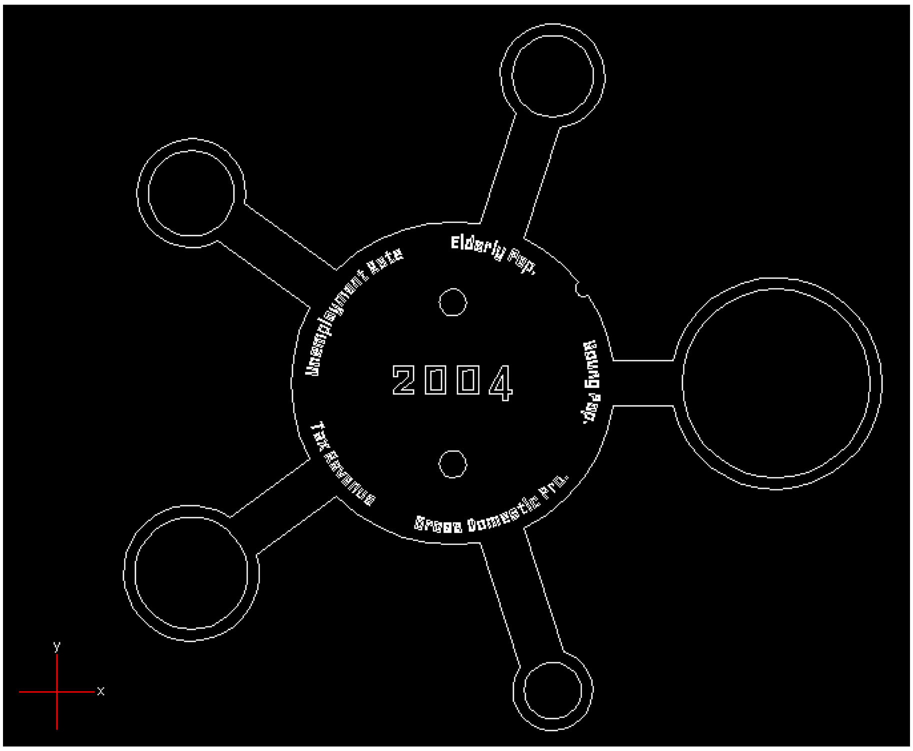


Figure 7.5: DXF file output illustrating one instance of the whole set

Figure 7.6 exhibits three stacked instances and a close-up shot revealing the details of the physical objects. In figure 7.7 photos of the stack of whole set is shown along with another close-up shot.

The requirements denoted by **R2** are satisfied by generating a set of DXF file with PhysVis and then manufacturing the tangible models with a laser cutter.


Figure 7.6: 3 Instances of Data Circles are stacked together


Figure 7.7: Whole set of Data Circles are stacked together

**R2.1: Demonstrate AR application on 'Data Circles' model**


Figure 7.8: Marker image installed on one instances of Data Circles model

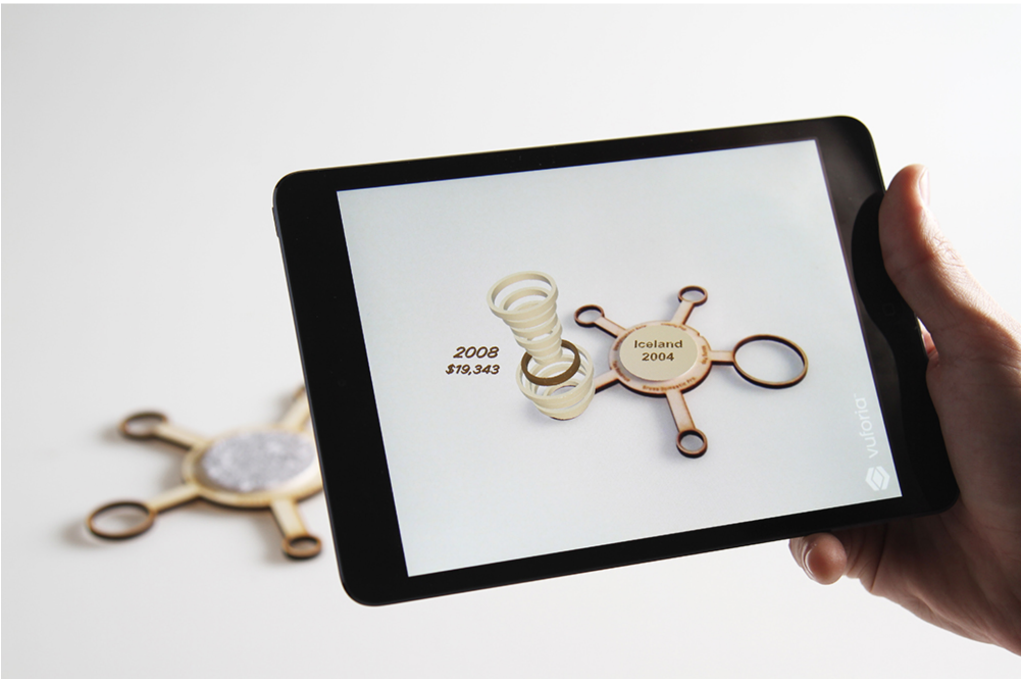


Figure 7.9: Demonstration of proposed AR interface on Data Circles model

In order to demonstrate the augmented reality application, the first instance of the model set is selected and an apparatus is designed to be 3D printed so that the marker image for tracking can be installed and removed with ease.

Figure 7.8 shows the photo of the model with marker image installed. Figure 7.9 shows the demonstration of the suggested interface in action.

Requirement denoted by **R2.1** is met by demonstration of the augmented reality interface, suggested in section 6.2.1, on one of the instance of the Data Circles model.

## 7.3 Discussion

In this chapter, validation of the physical visualization pipeline, called PhysVis, was made by meeting the specified requirements in the related section. A CSV file is generated in order to store the obtained data from OECD and PhysVis is utilized for generating DXF and STL outputs for two different physical visualization models. The key role of PhysVis is to automate the whole process with predefined models while offering full flexibility for user to adjust almost each parameter defining the models. The use of PhysVis for generating outputs ready to be fast prototyped with either 3D printing or laser cutting technologies is followed by a manufacturing process, which took relatively short amount of time to deliver the results. Those technologies are fast and affordable, therefore contributes to the feasibility of physical visualizations and supports the practicality of PhysVis.

Validation of the pipeline prompts probable use cases of physical visualizations in various settings. For example, in a business environment, use of common types of on-screen visualizations are not interesting hence the likelihood of less impact on audiences. Introduction of PhysVis pipeline with an affordable fabrication infrastructure could facilitate the use of physical visualizations in business meetings and thus induce more interactivity among participants. Engaging nature of physical visualizations is a proven fact shown by conducted studies and effective use of their potential can change the way we interact with data in real-life scenarios.

On top of that, utilization of augmented reality technology through physical visualization objects further expands the probable use scenarios by introducing supplementary features and more flexibility. Physical objects with augmented information displayed on top of them can reach the potential of on-screen visualizations yet retain the physicality. For example, a classified information can be embedded into a tangible object to be safely hid. With the help of a dedicated augmented reality application the information could be revealed, given that, 3D object tracking is possible with augmented reality applications as explained in section 3.5. This would effectively eliminate the need of a tracker image on the object, which could hint the object and compromise it in the case of given example.

Chapter 8

# Conclusion and Future Work

## 8.1 Conclusion

The research conducted throughout the thesis work revealed that data physicalization is a rather novel yet quickly growing research topic as a subset of InfoVis and HCI. The advent of rapid prototyping technologies has a substantial impact on the viability of the subject by enabling quick and accessible manufacturing. Given that industrial manufacturing processes are expensive and not easy to access, conducting research to gain better insight into the subject and reveal its potential is difficult due to the fact that the manufacturing process is a crucial and indispensable step to validating the feasibility of physical visualizations. On top of that, accessible enabling technologies play an important role for physical visualizations to become widely available in daily life settings.

In the related chapter, it is illustrated that physical visualizations are highly promising. Their physical form is a compelling factor for attracting further attention of the one interacting with the tangible object. It is declared that physical objects have several positive influences on people when telling stories or showing trends are concerned.

Given the fact that physical visualizations have proven cognitive advantages over traditional visualization methods, the lack of design research on physical visualization models is one factor limiting their potential. Furthermore, generating digital design files to physically realize those models is another overwhelming task to be achieved.

Those problems are first identified and then a solution is proposed within this thesis work. The most significant contribution of the thesis is the introduction of data physicalization pipeline called PhysVis. This tool not only facilitates the process of generating digital files to be manufactured but also introduces two novel designs to physical visualization model family.

The purposeful enhancement, outside artistic intention, of physical data objects through augmented reality, as demonstrated in this thesis work, is the first attempt in literature. It is accepted that physical visualizations have some limitations compared to digital visualizations, whereas it is validated that AR technology works properly with generated tangibles to introduce more features and to attempt to close this gap.

## 8.2 Future Work

This section offers further advices on any future attempts at contributing to the conducted research.

The pipeline supports all steps for realizing tangible models. However, the augmented reality software is not included within the pipeline due to the use of Vuforia platform with the intention of quick, proof-of-concept solution. However, there are JavaScript library solutions available. Utilization of those libraries can pave the way for the inclusion of the AR interface into the full pipeline.

The file format for the 3D models generated by PhysVis is defined as STL. A more advanced alternative file format is VRML, which can also specify color information for each of the surfaces of the generated mesh as explained in subsection 3.2.2. Support of VRML by PhysVis can enable the generation of fused Data Tower models with distinctive colors. It should be noted that VRML file format only works with compatible 3D printers with multiple extruders.

Physical visualizations are the first step to embedding data into tangible objects. However, those static objects are bound to the information they are meant to represent. Again, this is another pitfall of *static* physical visualizations in comparison to digital visualizations. Dynamic physical visualization is another emerging research topic that requires much more attention to introduce more practicality to tangible visualizations.

Use of dynamic physical visualizations would also eliminate the need for repetitive manufacturing for each data set. However, controlling a dynamic system also requires a software product for handling various tasks, such as data acquisition, programming various modes and et cetera. Considering these requirements, a software product could also be implemented to accompany dynamic physical visualizations.

# Bibliography

[1] M. Friendly, "A brief history of data visualization," in *Handbook of data visualization.* Springer, 2008, pp. 15–56.

[2] S. Swaminathan, C. Shi, Y. Jansen, P. Dragicevic, L. A. Oehlberg, and J.-D. Fekete, "Supporting the design and fabrication of physical visualizations," in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems.* ACM, 2014, pp. 3845–3854.

[3] Y. Jansen, P. Dragicevic, and J.-D. Fekete, "Evaluating the efficiency of physical visualizations," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, 2013, pp. 2593–2602.

[4] R. A. Khot, L. Hjorth, and F. Mueller, "Understanding physical activity through 3d printed material artifacts," in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems.* ACM, 2014, pp. 3835–3844.

[5] (2017) Data visualization 101: How to design charts and graphs. [Online]. Available: https://visage.co/content/data-visualization-101/

[6] Brosvision. Augmented reality marker generator. [Online]. Available: http://www.brosvision.com/ar-marker-generator/

[7] M. Aparicio and C. J. Costa, "Data visualization," *Communication design quarterly review*, vol. 3, no. 1, pp. 7–11, 2015.

[8] R. E. Chaddock and W. C. Brinton, "Graphic methods for presenting facts." 1915.

[9] D. Andrews, E. Fowlkes, and P. Tukey, "Some approaches to interactive statistical graphics," *Dynamic Graphics for Statistics*, pp. 73–90, 1988.

[10] J. R. Beniger and D. L. Robyn, "Quantitative graphics in statistics: A brief history," *The American Statistician*, vol. 32, no. 1, pp. 1–11, 1978.

[11] D. C. Sircar, *Studies in the geography of ancient and medieval India.* Motilal Banarsidass Publ., 1990, vol. 11.

[12] E. R. Tufte and E. Weise Moeller, *Visual explanations: images and quantities, evidence and narrative.* Graphics Press Cheshire, CT, 1997, vol. 36.

[13] M. Friendly and E. Kwan, "Effect ordering for data displays," *Computational statistics & data analysis*, vol. 43, no. 4, pp. 509–539, 2003.

[14] J. M. Vaquero and M. Vázquez, *The Sun Recorded Through History.* Springer Science & Business Media, 2009, vol. 361.

[15] W. Playfair, H. Wainer, and I. Spence, *Playfair's commercial and political atlas and statistical breviary.* Cambridge University Press, 2005.

[16] W. Playfair, *The commercial and political atlas: representing, by means of stained copper-plate charts, the progress of the commerce, revenues, expenditure and debts of england during the whole of the eighteenth century.* T. Burton, 1801.

[17] S. Few, "Data visualization for human perception," *The Encyclopedia of Human-Computer Interaction, 2nd Ed.*, 2013.

[18] E. R. Tufte and P. Graves-Morris, *The visual display of quantitative information.* Graphics press Cheshire, CT, 1983, vol. 2, no. 9.

[19] J. Bertin, *Semiology of graphics: diagrams, networks, maps.* University of Wisconsin press, 1983.

[20] Y. Jansen, "Physical and tangible information visualization," Ph.D. dissertation, Citeseer, 2014.

[21] B. Lee, P. Isenberg, N. H. Riche, and S. Carpendale, "Beyond mouse and keyboard: Expanding design considerations for information visualization interactions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2689–2698, 2012.

[22] C. S. Henshilwood, F. d'Errico, R. Yates, Z. Jacobs, C. Tribolo, G. A. Duller, N. Mercier, J. C. Sealy, H. Valladas, I. Watts *et al.*, "Emergence of modern human behavior: Middle stone age engravings from south africa," *Science*, vol. 295, no. 5558, pp. 1278–1280, 2002.

[23] C. Cain, "Implications of the marked artifacts of the middle stone age of africa," *Current Anthropology*, vol. 47, no. 4, pp. 675–681, 2006.

[24] A. K. Bangura, *African mathematics: From bones to computers.* University Press of America, 2012.

[25] V. Pletser and D. Huylebrouck, "The ishango artefact: The missing base 12 link," *FORMA-TOKYO-*, vol. 14, no. 4, pp. 339–346, 1999.

[26] D. Schmandt-Besserat, *How writing came about.* University of Texas Press, 2010.

[27] S. Swaminathan, C. Shi, Y. Jansen, P. Dragicevic, L. Oehlberg, and J.-D. Fekete, "Creating physical visualizations with makervis," in *CHI'14 Extended Abstracts on Human Factors in Computing Systems.* ACM, 2014, pp. 543–546.

[28] I. Gwilt, A. Yoxall, K. Sano *et al.*, "Enhancing the understanding of statistical data through the creation of physical objects," in *DS 73-1 Proceedings of the 2nd International Conference on Design Creativity Volume 1*, 2012.

[29] M. A. Borkin, A. A. Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister, "What makes a visualization memorable?" *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2306–2315, 2013.

[30] P. Isola, D. Parikh, A. Torralba, and A. Oliva, "Understanding the intrinsic memorability of images," in *Advances in Neural Information Processing Systems*, 2011, pp. 2429–2437.

[31] S. M. Kosslyn, "Understanding charts and graphs," *Applied cognitive psychology*, vol. 3, no. 3, pp. 185–225, 1989.

[32] W. S. Cleveland and R. McGill, "Graphical perception: Theory, experimentation, and application to the development of graphical methods," *Journal of the American statistical association*, vol. 79, no. 387, pp. 531–554, 1984.

[33] S. Stusak, M. Hobe, and A. Butz, "If your mind can grasp it, your hands will help," in *Proceedings of the TEI'16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM, 2016, pp. 92–99.

[34] S. Stusak, J. Schwarz, and A. Butz, "Evaluating the memorability of physical visualizations," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 3247–3250.

[35] P. Marshall, "Do tangible interfaces enhance learning?" in *Proceedings of the 1st international conference on Tangible and embedded interaction*. ACM, 2007, pp. 163–170.

[36] E. A. Locke and G. P. Latham, *A theory of goal setting & task performance.* Prentice-Hall, Inc, 1990.

[37] S. Stusak, A. Tabard, F. Sauka, R. A. Khot, and A. Butz, "Activity sculptures: Exploring the impact of physical visualizations on running activity," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 2201–2210, 2014.

[38] S. Stusak and A. Aslan, "Beyond physical bar charts: an exploration of designing physical visualizations," in *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2014, pp. 1381–1386.

[39] S. J. Szigeti, A. Stevens, R. Tu, A. Jofre, A. Gebhardt, F. Chevalier, J. Lee, and S. L. Diamond, "Output to input: concepts for physical data representations and tactile user interfaces," in *Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems*. ACM, 2014, pp. 1813–1818.

[40] D. R. Nadeau and M. J. Bailey, “Visualizing volume data using physical models,” in *Visualization 2000. Proceedings.* IEEE, 2000, pp. 497–500.

[41] N. Gershenfeld, “How to make almost anything: The digital fabrication revolution,” *Foreign Aff.*, vol. 91, p. 43, 2012.

[42] E. Canessa, C. Fonda, M. Zennaro, and N. DEADLINE, “Low–cost 3d printing for science, education and sustainable development,” *Low-Cost 3D Printing*, vol. 11, 2013.

[43] H. Lipson and M. Kurman, *Fabricated: The new world of 3D printing.* John Wiley & Sons, 2013.

[44] C. Mota, “The rise of personal fabrication,” in *Proceedings of the 8th ACM conference on Creativity and cognition.* ACM, 2011, pp. 279–288.

[45] L. Roscoe *et al.*, “Stereolithography interface specification,” *America-3D Systems Inc*, 1988.

[46] M. Burns, “The stl format: standard data format for fabbers,” *Automated Fabrication*, 1993.

[47] M. Bar and M. Neta, “Humans prefer curved visual objects,” *Psychological science*, vol. 17, no. 8, pp. 645–648, 2006.

[48] R. L. Harris, *Information graphics: A comprehensive illustrated reference.* Oxford University Press, 2000.

[49] M. Friendly and S. S. Course, “Categorical data analysis with graphics,” *SCS Short Course*, 2003.

[50] J. Steele and N. Iliinsky, *Beautiful visualization: looking at data through the eyes of experts.* ” O’Reilly Media, Inc.”, 2010.

[51] A. V. Moere, M. Tomitsch, C. Wimmer, B. Christoph, and T. Grechenig, “Evaluating the effect of style in information visualization,” *IEEE transactions on visualization and computer graphics*, vol. 18, no. 12, pp. 2739–2748, 2012.

[52] S. Bateman, R. L. Mandryk, C. Gutwin, A. Genest, D. McDine, and C. Brooks, "Useful junk?: the effects of visual embellishment on comprehension and memorability of charts," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, 2010, pp. 2573–2582.

[53] A. Gillet, M. Sanner, D. Stoffler, and A. Olson, "Tangible interfaces for structural molecular biology," *Structure*, vol. 13, no. 3, pp. 483–491, 2005.

[54] R. Sluis, I. Weevers, C. Van Schijndel, L. Kolos-Mazuryk, S. Fitrianie, and J. Martens, "Read-it: five-to-seven-year-old children learn to read in a tabletop environment," in *Proceedings of the 2004 conference on Interaction design and children: building a community.* ACM, 2004, pp. 73–80.

[55] O. Zuckerman, S. Arida, and M. Resnick, "Extending tangible interfaces for education: digital montessori-inspired manipulatives," in *Proceedings of the SIGCHI conference on Human factors in computing systems.* ACM, 2005, pp. 859–868.

[56] L. Terrenghi, M. Kranz, P. Holleis, and A. Schmidt, "A cube to learn: a tangible user interface for the design of a learning appliance," *Personal and Ubiquitous Computing*, vol. 10, no. 2-3, pp. 153–158, 2006.

[57] K. Lubick. Stlexporter. [Online]. Available: https://github.com/mrdoob/three.js/blob/a72347515fa34e892f7a9bfa66a34fdc0df55954/examples/js/exporters/STLExporter.js

[58] E. Grey. Filesaver.js. [Online]. Available: https://github.com/eligrey/FileSaver.js/

[59] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Visual Languages, 1996. Proceedings., IEEE Symposium on.* IEEE, 1996, pp. 336–343.