Delft University of Technology
Master of Science Thesis in Embedded Systems

Designing Safe 2.5D Tangible Shape Displays

Kanya Satis
Supervisor: Dr.Mitra Nasri

Embedded Networked Systems

TU Delft Delft University of Technology

Designing Safe 2.5D Tangible Shape Displays

Master of Science Thesis in Embedded Systems

Embedded and Networked Systems Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands

Kanya Satis
k.satis@student.tudelft.nl
skanya93@gmail.com

Supervisor: Dr.Mitra Nasri
m.nasri@tue.nl

14th September 2020

Author
Kanya Satis (k.satis@student.tudelft.nl)
(skanya93@gmail.com)
Title
Designing Safe 2.5D Tangible Shape Displays
MSc Presentation Date
30th September 2020

Graduation Committee

Dr. R. Venkatesha Prasad,	Delft University of Technology
Dr. Mitra Nasri	Eindhoven University of Technology
Dr. ir. Arjan Van Genderen	Delft University of Technology

Abstract

2.5D shape display is a recent idea in the market that emerged as a platform of interaction between a computer and human. 2.5D shape display is essentially a grid like matrix consisting of actuators and pins moving up and down in vertical motion to create pseudo 3D images. Focused as a visual display in some applications and as a medium of input or output in others, this technology holds a lot of potential to be explored in present and future applications since it is a unique type of hardware that can actually show images in a real world. Recently, many such platforms have been created by the use of different hardware solutions and have found applications in gaming interface, physical tele-presence, dynamic objects etc. Some research areas that can still be explored are applications for the blind, building and object modelling etc.

With advances in applications, there is a need for hardware with higher resolution at lower cost. Also, being an interact-able display, safety and comfort of the user is of utmost importance. This has been seen to be lacking in the existing projects. In this thesis, we focus on designing a prototype for building safe display with minimal cost. We then go on to understand safety and comfort for the chosen display prototype and design some safe actuation algorithms. These are later evaluated using a combination of an experimental survey and simulation to find and propose a good solution for safe 2.5D shape displays.

Preface

The last few months since the start of my thesis has been an amazing uphill journey. A lot of sleepless nights with debugs and analysis, but a very beautiful journey overall. The entire journey was very challenging and educational, exactly what I was looking for as I started my thesis. This thesis turned into a reality with the help and support of many individuals. So, at the outset of this report, I would like to extend my immense gratitude toward everyone who has helped me in this endeavor.

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Mitra Nasri for introducing me to an amazing field and for the continuous support, guidance and motivation. Her unwavering enthusiasm kept me completely engaged in my research work. I extend my profound thanks to my master thesis committee members Dr. Ranga Rao Venkatesha Prasad and Dr. Ir. Arjan Van Genderen.

Next, I would like to thank all the participants of my survey for taking time out of their busy schedule and making arrangements in spite of the covid-19 regulations to actively give feedback and comments on the experiments. I would also like to thank Rajya for proof reading numerous versions of my thesis report and giving me critical feedbacks.

Delft became my home due to the presence of some beautiful souls. For this, I would like to thank Anil Kumar Kumaran, Kriti Dhingra, Pragati Kidambi, Ashwita Nair, Pranjal Singh Rajput and Sonnya Dellarosa. I would also like to thank my friend Sri Elavarasan for being my constant support.

Finally, I would like to thank my family - my mother, father, sister and grandmother, for always believing in me and supporting me in my toughest of times. All their hard works are reflected in the results of this work.

Kanya Satis

Delft, The Netherlands
16th September 2020

Contents

Chapter 1

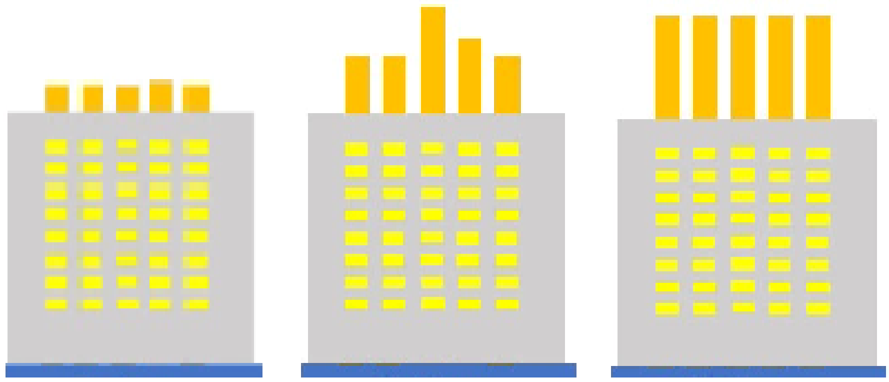
Introduction

Shape perception is an important part of our everyday experience, and still the idea of an interactive shape display remains a huge challenge. A hardware with a perfect combination of actuation, safe tangibility and easy configuration can become a revolutionizing interface. Such a hardware with rich perception can be a great display for people with visual impairments. This can also create a wide range of applications in virtual reality.

An ideal tactile interface is a human-machine interface that provide a physical material which couples the digital world to the physical world. A simple example is the computer mouse. As we move the mouse in the physical world, the pointer in the screen traces the hand motion, thus creating a link between the two worlds. Though there are many such interfaces developed, they are limited by their physical attributes and leads to interfaces that cannot exactly represent all the necessary information. A shape changing device tries to overcome this setback by providing a platform which can change its shape, size, and orientation thus bringing the digital interface into the physical world. One of the state-of-the-art type of interface aimed for this is the 2.5D shape display. Consisting of a grid like arrangement, 2.5D shape displays have equally spaced actuators capable of vertical motion. These actuators when placed at different vertical positions, create a semi 3-dimensional structure, thus giving it the name 2.5D model. Figure 1.1 shows how these pin actuations will look like.

Research in the area of tactile shape changing devices have been growing over the years. The majority of the research groups in this field have been working on the hardware interface, on how to make the actuators smaller to increase the resolution [27],[9], how to make the actuators withstand the pressure while not compromising on the resolution [19], interfacing the real world scenarios to the hardware[15], [26], or development of advanced applications limited to the currently available prototypes of the display[25]. There are very few researches on the user experience side [8], where they are limited to how the texture feel is provided or to develop haptic feedback systems. While all these are major challenges in this field, one of the most important aspect, the safety and comfort of users has not been considered in most of the research scenarios that we came across.

There have been multiple applications developed based on 2.5D shape displays. Some of these are physical tele-presence, city-scape design, geographical



(a) **Pins at lowest position** (b) **Pins creating an image** (c) **Pins at highest position**

Figure 1.1: **2.5D Shape display - Pin motion**

modelling, gaming, creating physical forms, braille displays etc.[16], [25], [17]. Each of these are created on different display having different resolution. What happens if we can increase the resolution further? When the resolution increases, most of the applications mentioned can be ported to this one display. This will create the application scenario with much better accuracy. In addition, it widens the scope by making it possible to create further applications such as guidance system or virtual view for the visually impaired, high definition gaming interface etc.

As the resolution of the displays increased, we observed that the cost increased as well. How can we design a cost-efficient display with a higher resolution? Figure 1.2 shows a placement of these applications in a resolution vs price plot. The red circle shows our area of focus for this thesis. Most of these applications need a tangible aspect. To make it tangible during actuation, the safety and comfort of the users needs to be considered. We need to ensure that the user will not be injured or experience any form of physical pain or discomfort when interacting with the display. With this as the base, the next session will explain the problem statement of this thesis and its motivation behind the same.

1.1 Problem Statement

For a human being, the haptic sense is composed of two main aspects[21] : the tactile sense and the kinesthetic sense. Tactile sense refers to the sense of touch and kinesthetic sense refers to the sense of force or motion. With these being the main two aspects of focus for any tactile interface, best user experience can be achieved by improving these experiences. Higher the resolution, better will be the tactile experience. Similarly, smoother transition will provide better kinesthetic experience.

From the earliest of the researches in the field of shape changing devices, resolution was the biggest challenge for many years. Since the 1990s, multiple researchers [27], [19], [9] have been devoted to create actuators with best resolutions possible. Some have been successful in their endeavors as well. But, these had their own drawbacks in terms of cost, refresh rates, maximum stroke

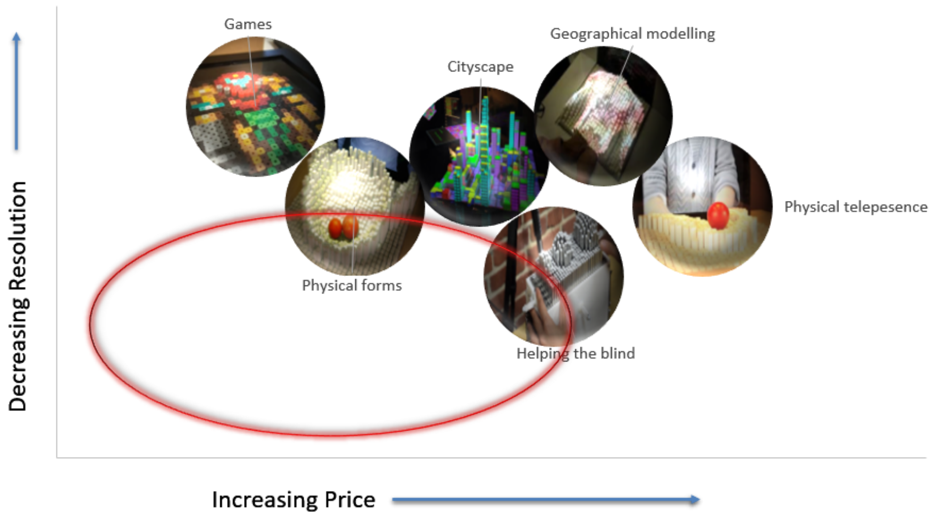


Figure 1.2: **Applications - Resolution vs price**

lengths, stalling pressure etc. This leads us to think what can be good trade off to ensure good resolution, without compromising much on the other areas.

It was seen that very few researches focus on the user experience, but tends to be more inclined towards how the user feels with the effect of resolution or haptic response and much lesser on the safety aspect of their applications. But, what can be deemed as a safe transition or what can be the best transition for a given application instance? Though the shape changing technology field has matured in the recent years, these questions are not an area of focus in most of the projects.

With these queries and objectives in mind, the following research objectives were created.

1. **How to build a 2.5D shape display with adequate trade-off between resolution and cost?**
2. **What can be deemed as safety and comfort for a 2.5D shape display? And how can this be achieved by better controlling the actuation when creating 2.5D images?**

The aforementioned objectives are coupled with the following challenges:

- There is no definite definition of *safe and comfort* in 2.5D displays. So, what is *safe and comfortable* needs to be defined and evaluated.
- There is no standard method to evaluate the algorithms that we design without an actual prototype at hand. This method of evaluation needs to be formulated.

1.2 Solution Approach

To meet the research objectives, in this thesis we design a three part approach. First, we will determine the hardware requirements and then design a 2.5D shape display based on these requirements. For this, we study a set of existing actuators, sensors and design structures to generate a design that satisfies the requirements.

Second, we define some safety and comfort guidelines for the chosen design. For this purpose, we create a small prototype and by the method of survey from few people, we conduct an initial study to understand the human interfacing experience. Based on this result, we extend the study to create specific scenarios that we replicate in the hardware. Some participants tested this experience to generate a *safety guideline* for the chosen design.

Finally, based on the initial safety study, we create a few actuation algorithms to control the movement of actuators. To understand its performance, we create a simulator which mimics the working of a 2.5D display and implemented the actuation algorithms in the simulator environment. Next, we evaluate these algorithms to understand how they perform against various parameters such as refresh rate, power efficiency and safety and comfort. We test this for static and dynamic image creation scenarios to conclude which *safe actuation algorithms* are best based on different scenarios and needs.

1.3 Contributions

The main contributions of this thesis are as follows:

1. The choosing of appropriate parts for the prototype design and coming up with a design to create a cost-efficient and high-resolution display.

2. Defining safety and comfort guidelines for the chosen design by prototyping, survey and analysis of the results.

3. Designing and implementation of different safe algorithms in a simulator environment that can be directly ported to the actual hardware.

4. Evaluation of the implemented algorithms to ensure safety and performance as per the required application instance.

1.3.1 Changes due to covid-19

The initial plan of the thesis was to build the complete hardware and test the safe actuation algorithms in the hardware. After the initial study phase, we approached Eindhoven University of Technology to fund the project. They approved the funding and offered to provide lab support to design and build the hardware. But, due to covid-19 measures, the university had to be closed. Because of this, the prototype could not be built, and we had to create a simulator setup that mimics the working of the 2.5D display which was used to test the algorithms.

1.4 Report organization

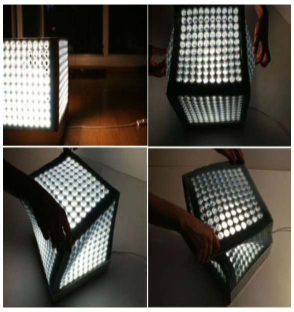
The report is organized in the following manner: Chapter 2 provides a brief description of 2.5D display and its related research. In Chapter 3, we study and compare various hardware components of the display to create the required design. Chapter 4 outlines the way a prototype of the design is created to study the safety aspect, addressing the second research objective. It also discusses the survey conducted and the results of the analysis to describe safety and comfort. Based on the results of the survey, we design six safe actuation algorithms which are discussed in chapter 5. Due to the unavailability of the hardware, we need to create a simulator and evaluate the performance of the designed algorithms. These are done in chapter 6. The conclusions of the research and possible future works planned are presented in Chapter 7.

Chapter 2

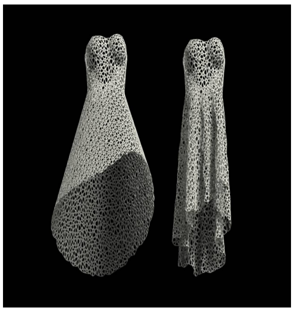
Background & Related Work

2.1 Shape Displays

Shape Displays are generally any technological interface or even objects which can change its own shape, texture or mould themselves to resemble other objects. This has found use in areas like a 3D-like interface, transformable day to day objects, or even gaming platforms.

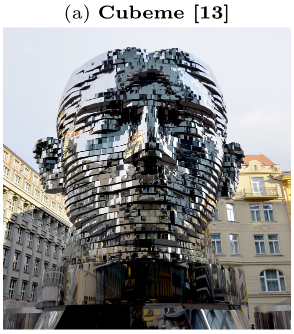


(a) **Cubeme [13]**



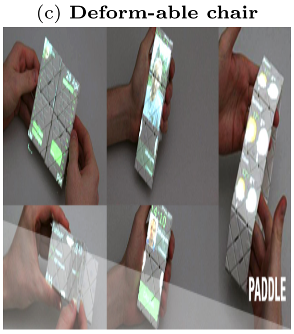
(b) **Shape Changing Dress**

(c) **Deform-able chair**



(d) **Head of Franz Kafka [28]**

(e) **3D Shape Display [1]**



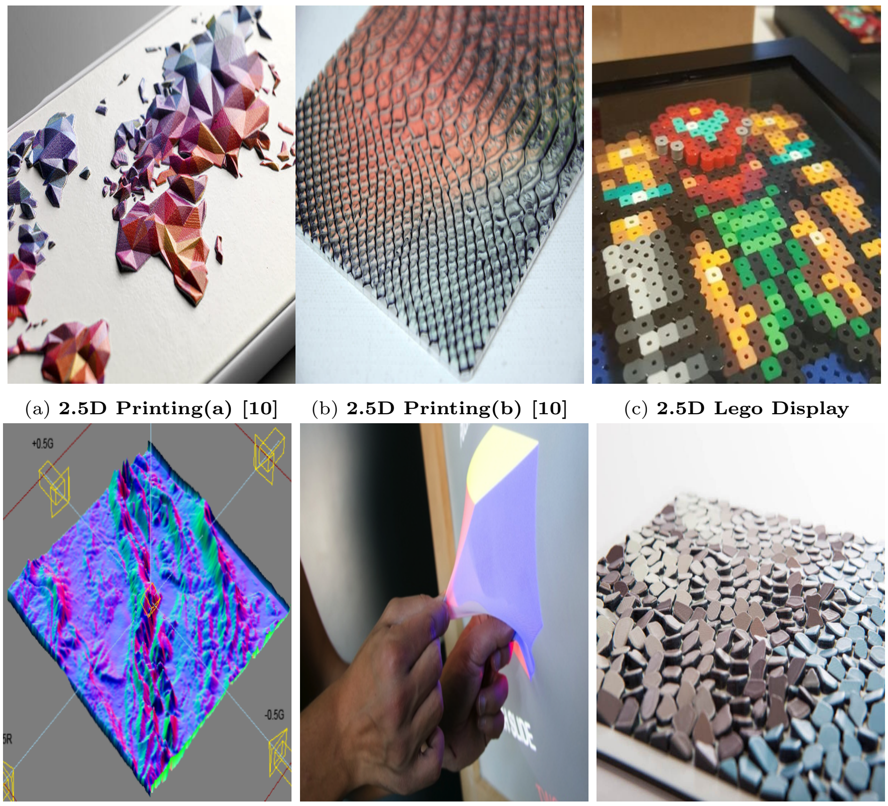
(f) **Paddle [23]**

Figure 2.1: **Shape Displays**

Shape displays vary from the simple shape changing lamp designed by Ewa Garniec [13] as in figure 2.1a to the famous shape changing statue of Franz Kafka built in Prague, Czech Republic by David Černý [28] as in figure 2.1d. Some other examples include the design changing dress designed by the 4D printing technology as in figure 2.1b and the *Breathing Chair* designed by Yu-Ying Wu as in figure 2.1c. There are also some visual display devices such as the 3D Shape Display [1] as in figure 2.1e and deform-able mobile phones called paddle [23] as in figure 2.1f.

2.2 2.5D Displays

Two and a half Dimensional (2.5D) perception, alternatively known as pseudo 3D is a 2D projection or like technique which is used to generate shapes and scenarios that resemble them in the 3D space that we live in. This can be of many forms. Initially started with computer graphics in movies and video games to get this effect, these have been developed into actual hardware devices that we can interact with.



(a) **2.5D Printing(a) [10]** (b) **2.5D Printing(b) [10]** (c) **2.5D Lego Display**

(d) **2.5D Landscape model** (e) **GHOST [5]** (f) **2.5D Printing(c)**

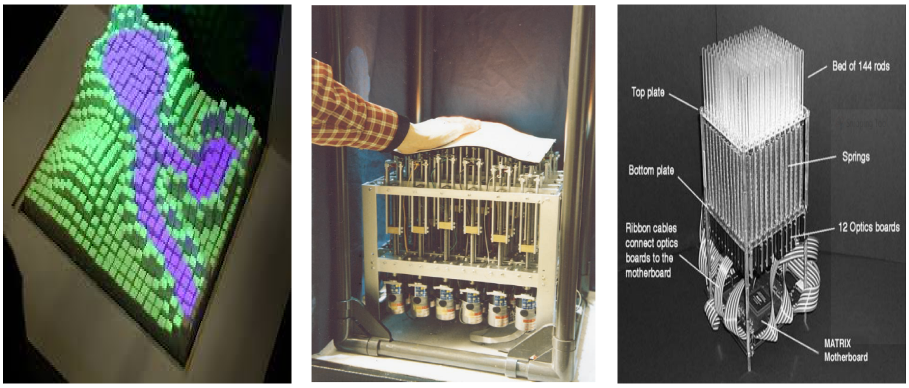
Figure 2.2: **2.5D Displays**

2.5D printing [10] was developed by researchers at the Eindhoven University of Technology. These were made to mimic shape, texture and feel into surfaces.

This is shown in figure 2.2a and 2.2f. Similar 2.5D printing can be seen in figure 2.2b where a raised LED UV direct to substrate is printed on aluminium. Generic, highly-organic shape-changing interfaces also known as GHOST [5] are displays made from malleable materials which can deform and change shape to generate 2.5D images as seen in figure 2.2e. Figure 2.2c shows a 2.5D lego display while figure 2.2d shows a light based 2.5D landscape model.

2.3 2.5D Shape Displays and Related works

Combining the idea of 2.5D displays and shape displays, we arrive at 2.5D Shape Displays. These are generally a grid of actuators that move vertically up and down to create different pseudo-3D images. The motivation of this project was from the **InFORM** [16], a tangible media group project which was developed as a prototype to research various applications in this field. Some of these applications include physical tele-presence, city-scape models, dynamic furniture etc.

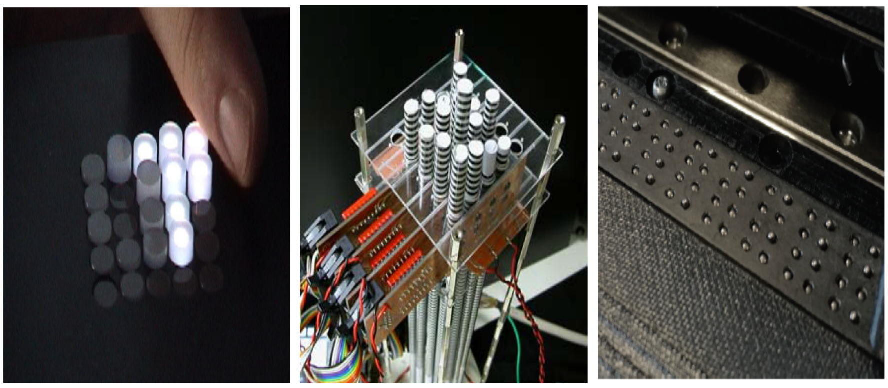


(a) **inFORM [16]** (b) **Feelex [14]** (c) **Matrix [20]**

Figure 2.3: **2.5D Shape Displays**

FEELEX: Developed in 1997, this was one of the earliest 2.5D shape displays created [14]. This had a 6×6 array of linear actuators to create a display of 4cm resolution. Each linear actuator is driven by a DC motor an a screw mechanism to convert to linear motion. A force sensor at the top is used to receive tactile feedback and an optical encoder connected to the DC motor is used to get positional feedback. Being one of the first prototypes created, these were of very low resolution and bulky, but this was used as a platform to study various hardware and user experience parameters.

Matrix: Multipurpose Array of Tactile Rods for Interactive eXpression [20] was a 2.5D tactile interface that was created to receive information rather than to perceive images. Unlike other prototypes this did not have a driving mechanism, but a set of sensors to read the positions the pins come to when hand pressure is applied. This has 144 pins in 12×12 grid with a resolution of about 1cm. A higher resolution was achieved due to the lack of driving mechanism. They use quadrature encoders for each pin to find the position which is fed to an FPGA to recreate the image electronically.



(a) **Lumen [22]** (b) **PopUp [19]** (c) **BrailleCursor [17]**

Figure 2.4: **2.5D Shape Displays**

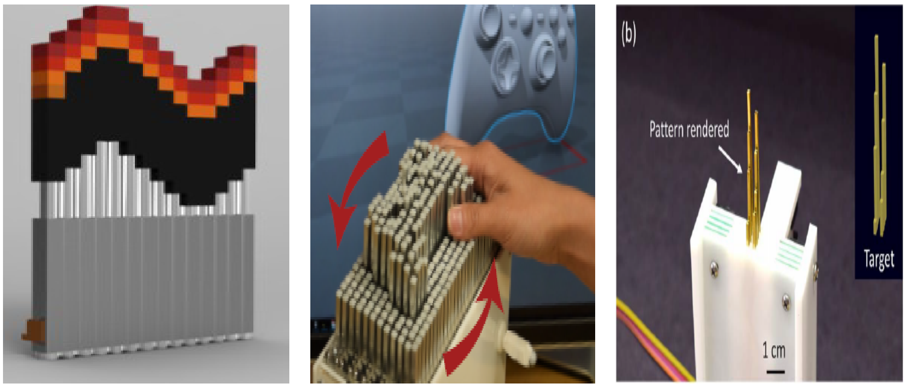
Lumen: An SMA(Shape Memory Alloy) actuator system is employed in Lumen [22]. The actuator is mounted by a light guide that can change its color and height. SMA strings work on the principle of rapid contraction on passing of current and relaxing to actual length as the heat produced reduces. Because of this, the actuation is smooth and noiseless. But it has it disadvantage in terms of refresh rate and maximum stroke length that can be achieved. It receives tactile feedback through a custom made Smart Skin sensor. This has its advantages over sensors that depend on light such as encoders or cameras.

PopUp: Like Lumen, PopUp also has SMA based actuation with a resolution of 1.2cm [19]. The SMA actuator used is Bio Metal Helix 200, which can increase to twice its length, and this being a very thin actuator of about 0.85mm in diameter, it could create high resolution displays. Tiny photo-reflectors were used to read the position of the actuators.

BrailleCursor: Braille displays are brilliant examples of 2.5D displays. The BrailleCursor was a refresh-able braille display built using electro-magnetic actuators and passive pins [17]. This had an excellent resolution of 0.4mm, but was constrained to the fact that it was a digital display where the pins could assume either a high or a low position only. It also had a very small stroke length of 3mm, though it served the application purpose, it will be hard to be perceived as a scalable device.

Zixel: Zixel is a design made with a combination of actuators and RGB cubes that help provide the visual and tactile feel of the display [9]. The actuation is based on piezo-electric linear drive motors which can create substantial speed and force. It has a very high resolution of 82dpi and a stroke length of about 2.8cm. Though it has one of the best resolutions achieved in 2.5D displays, the hardware build around for the control and sensing of each actuator makes the prototype too bulky and inefficient in terms of energy. This is still a concept and the prototype is yet to be made.

ShapeShift: These use dc motors with lead screws to create linear actuation [25]. Photointerruptors are used as sensors to detect the quadrature markings on the pins. They have a resolution of about 7mm and a stroke length of 5cm. This design is very similar to the prototype designed as a result of the research, but ShapeShift has a better resolution.



(a) **Zixel [9]** (b) **ShapeShift[25]** (c) **ESB-Based display[27]**

Figure 2.5: **2.5D Shape Displays**

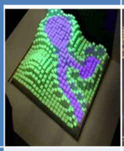
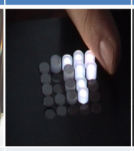
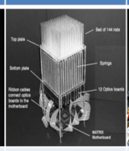
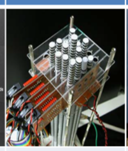
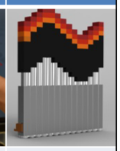
	Tangible	Feelex	Lumen	Matrix	PopUp	Shape-Shift	Zixel
Hardware							
1x1cm							
Resolution (Size of each pin)	2.5cmx2.5cm	4cmx4cm	~1cmx1cm	~1.3cmx1.3cm	0.7cmx0.7cm	0.7cmx0.7cm	0.3cmx0.3cm
Actuation	Motorized slide potentiometer	DC motor	SMA	Springs	SMA	DC motor	Piezoelectric linear drive motors
Sensors	A/D converter	Optical Encoder	Custom Smart Skin sensor	Quadrature encoder	Photo reflectors	Photo-interrupters	-

Figure 2.6: **Comparison of state-of-the-art displays**

ESB Based display: In this paper [27], the authors create an actuator aimed for designing very high resolution and compact 2.5D displays. Instead of linear actuators, they use high voltage electro-static brakes to hold the pin position. These can create a display of resolution as high a 1.7mm and a stroke length of 3.5cm. They are also very low cost but this actuator is not studied in this research as they are not in production.

2.3.1 Comparison of state-of-the-art displays

In the previous section we saw many displays and actuators created for 2.5D shape display. In this chapter, we will provide a summary of these to understand its drawbacks. Figure 2.6 shows a comparison table based on the resolution and the components used.

The table provides a clear understanding on the resolution of the displays. We can see that *zixel* has the best resolution available. This is a project in progress, with the idea for a possible design published. These displays are high cost in comparison to the others we saw due to the use of piezo-electric linear actuator. For creating a 3 cm$\times$3 cm display, this will need 100 actuators which increase the cost significantly. The next best resolution is for PopUp and ShapeShift. The PopUp uses an SMA actuator which is one of the most costliest option in the market. This will be discussed in chapter 3. The other displays have much lower resolution than what we aspire to design, as will be discussed in chapter 3.

From the study of these state-of-the-art displays, we can see that very few displays have high resolution, and those that have high resolution are not cost efficient. This is a major drawback, thus requiring us to build a new hardware.

Chapter 3

Hardware Design

This chapter discusses the hardware modules and the various options for each module. We look into the advantages and disadvantages of each option to choose the best option based on cost, size, time efficiency, market availability and robustness etc. We design a good prototype based on the aspects mentioned and provide a cost estimate for building the product as well.

In the next section, we will discuss the requirements of the design and why we have these particular requirements.

3.1 Design requirements

The efficiency of the system will depend on the criteria that we discuss below:

Price: We aim to build a cost-efficient system. So, the individual pin cost has to be as low as possible. This will depend on the cost of the actuator and sensor that we choose.

Size: The size of the actuator decides the resolution of the display. In our system, we aim to make the resolution better than most of the state-of-the art displays. We aim for a resolution of about 3 to 6 mm. With this resolution, most of the existing applications such as city-scape modelling, physical tele-presence and gaming can be implemented with much better accuracy. In addition, this will increase the scope of applications where accuracy or pin-pin to distance is of importance. One such application is to create a dynamic display for the visually challenged. Increasing the resolution higher than this will mean the number of pins will increase, which in turn will increase the cost.

Speed: The speed at which the actuator is moving decides the refresh rate of the display. But, this is a tangible display, meaning the user can interact via touch. When the pins move at high speed, this can cause discomfort, or even injury which requires us to limit the speed as well. So, speed in the range of 10-15 mm/s is desired because the speed can still be adjusted for the safety of the user.

Stroke length: The stoke length of the display decides the applications that the display can be used for. If this value is too small, then the applications are limited. With higher stroke length, the scope of the display expands. This means that with higher control accuracy, the display can be used for applications that need a few millimeters of stroke length or for those that need a few centimeters of stroke length. So, we prefer higher stroke length to expand our application scope.

Robustness: Since the user will be able to directly interact with the actuators, they should be robust. They should have a high stalling force that can withstand the actuator moving backwards due to the weight and pressure exerted by the hand.

Based on the discussed criteria, we need to choose the different modules. These will depend mainly on the actuators and sensors we choose. So, we explore these modules in detail in the next section to finalize them.

3.2 Mechanical modules

Any 2.5D shape display consists mainly of pins, actuators, sensors, and control unit. The pins are the dynamic part of the system which move up and down to create the display image. These pins need to be coupled with an actuator, which is the mechanism that enables the pin to move based on the control signals. The position of these pins need to be tracked, and this is where sensors are used. All these actuators need to be controlled based on the desired position and the sensed position. We will add the control logic to a control unit which we choose, that will also control the actuators and get feedback from the sensors. The basic interaction block between these modules is shown in block diagram figure 3.1.

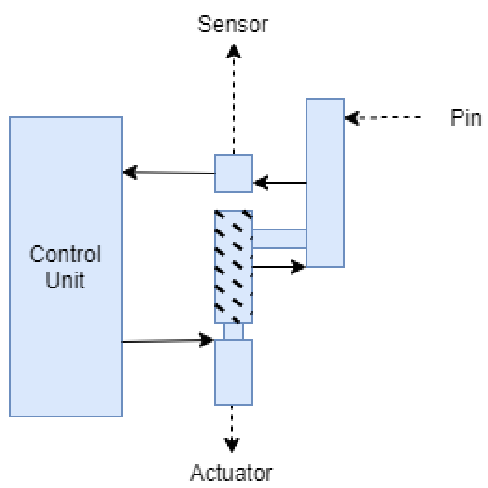


Figure 3.1: **System components**

3.2.1 Actuators

Actuator is the part of the system which is responsible for controlling and moving any mechanical part in a system. This requires a control signal and an energy source to create movement and this control signal can be of different type depending on the system. There are different types of actuators depending on the type of motion it provides. Since the 2.5D display needs actuators with one degree of motion that lets it move vertically, the actuator chosen for this purpose is a simple linear actuator.

For the 2.5D shape display, the main component of the system is the linear actuator. This decides the performance, efficiency and resolution of the system that can be built. So, this is the first component that was studied and chosen. All the other component decisions were made based on the actuator chosen.

There are different types of linear actuators based on source of motion. For this project a variety of such linear actuators are chosen and studied. From each type of linear actuator, a few specific actuators are chosen which can provide the high range of resolution. The major characteristics of these actuators such as the size, speed, force, robustness and cost are studied. After a comparative study, the most suitable option available is chosen.

Pneumatic actuators

Pneumatic, as the name refers to, is an actuator controlled using air pressure changes. The pneumatic actuator mainly has a piston and a cylinder, the piston being the moving part and the cylinder is where the compressed air is pushed into to create motion. These can create high force in a short time and it is very robust as the pressure can withstand high force applied on the piston. The control signal is the air pressure which determines the speed and distance the piston travels.

After a research into the available pneumatic cylinders, CJP Series pin air Cylinder from *Yueqing Hengxin Pneumatic Co., Ltd.* shown in figure 3.2 was chosen for the study. This has a diameter of 3 to 5 mm, satisfying the needed high resolution constraint and has a stroke length of 3 cm.

Figure 3.2: **CJP Pneumatic actuator[3]**

Though on a higher level, these seem to satisfy the criteria, there are other factors to consider. The control signal for pneumatic cylinders is compressed air, meaning the system will need to include an air compressor and a valve for each actuator which increases the cost, size and weight of the system. Air compressors are generally bulky and the best ones that fit into the needs of the system are those used in robotics. They offer the least weight possible, and still most of them weigh a minimum of 1kg. They are also too noisy and will make the system less portable. This is a major drawback of this actuator.

Hydraulic actuators

Hydraulic actuators are very similar to pneumatic actuators. They also comprise a piston and a cylinder mechanism, but the control signal used is pressurised liquid. Liquid being hard to compress can create a high force. But, this advantage is overshadowed by their major drawbacks for this application. They are mostly bulky, meaning they cannot give high resolution. They need a storage unit to contain the liquid and a motor system to circulate the liquid. They also emit a lot of heat, which requires a cooling system as well. All these make the system too bulky for the display application.

Shape Memory Alloy(SMA) actuators

These are thermal actuators that can be controlled by changing the temperature by Joule's effect. When electricity is passed through SMA material, this produces heat which contracts the material. When cooled, it expands creating the required actuation. These actuators are very compact and light weight. But the rate at which it can change the size is too slow for its use in dynamic displays. Also, precise control of these actuators is hard as temperature is also influenced by environmental changes. For the purpose of this study, we choose Biometal Helix BMX5020. This is a micro-coil which contracts when current is passed through it. It can expand up to 200% of its actual length at room temperature. It has a diameter of 0.2 mm, making it the best option for flexible high resolution display. But, the price of these materials is very high and it is not robust. A small pressure applied will bend the wire, distorting the system.

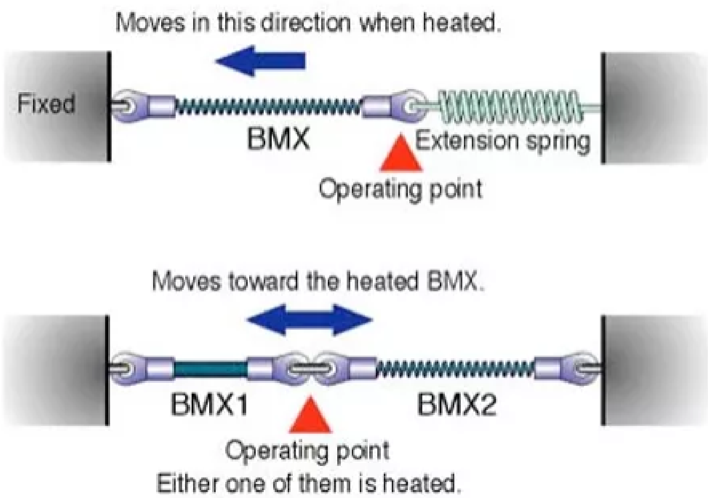


Figure 3.3: **SMA actuator working[2]**

Piezo-electric actuators

These devices can produce a high force movement with the application of a small voltage. They are mainly used for high precision displacement. It also has a high response time providing a high refresh rate. Multiple such devices can be stacked together to get bigger displacement length. But the displacement of a single actuator is in the order of few micro-meters. Stacking multiple actuators could provide utmost a few millimeters of displacement. This could be a great secondary actuator that can be used to get the precision, but as a primary actuator, it becomes a costly option due to the multiple number of actuators that will need to be stacked to get a good displacement.

Linear actuator

Linear actuators are essentially a set of motors, motor controllers, and pins coupled together to create a whole pin system. These are very easy to control and use, but the design of how these modules are packed cannot be changed making them bulky. Also, the cost of such linear actuators is much higher than its components taken separately. For the study, the VS-19 Pico Linear Servo by *Solarbotics* is considered. It can give a resolution of 30 mm when stacked side-by-side or up to 20 mm if stacked in two rows. They are very robust as they can handle a stalling torque of more than 60 gf.cm. It has a stroke length of 2 cm.

Geared system

A geared system to create a linear actuator is a set of motors with screws and gears that can be combined to produce a compact arrangement. The best gear system that can convert a rotary motion to a linear motion is a combination of a lead screw and a lead nut. The lead screw will be coupled to the shaft of the motor while the lead nut will be coupled to the lead screw and the pin. This will require more mechanical design and the system resolution will be limited by the size of the motor chosen.

For the study, three sets of motors were chosen. First is the DC motor TGPP06 with a planetary gear arrangement by TT motors. This has a diameter of 6 mm, but with this type of gear arrangement, resolution of up to 3 mm can be obtained by stacking. The second is the screw rod stepper motor LB0959. This is coupled with a lead screw and a slider. This can be used to design a system with a resolution of 5 mm. The third is a precision stepper motor with a diameter of 6 mm. This can be coupled with a worm gear or a lead screw to create a translating motion.

Comparison of actuators

The table 3.1 shows a comparison between the chosen actuators.

Actuator type	Price/pin (€)	Size(mm)	Speed(mm/s)	Stroke length(cm)
Pneumatic actuator	∼1	5	∼100	2
Hydraulic actuator	∼2	8-10	∼100	∼2-5
SMA actuator	30	0.05	10-20	2-3
Piezo-electric actuator	∼3	5-10	>200	<1
Linear actuator	10	30	12	2
Geared actuator system	∼1	∼5	10	4

Table 3.1: **Comparison of actuators**

(a) **DC motor with planetary gears[18]**

(b) **Stepper motor with lead screw[6]**

Figure 3.4: **Geared actuator system**

Based on the comparison table, it is clear that hydraulic actuators do not have any clear advantage over others. Pneumatic actuators though are cheap for the pin cost, an additional cost and size is added for the air compressor. SMA actuators while providing the best resolution, are too costly for the application. Piezo-electric actuators are mainly used for micro-precise-actuation. Their stroke length is in orders of micro meters, or when stacked up can give a few millimeters. These can be used along with another actuator to create accurate actuation, but cannot be used as a stand alone actuator. The geared actuator system offers everything the linear actuator provides with better cost and resolution. From these parameters, the closest to the required specification is a geared actuator system.

Of the two options considered for geared actuator system, one is a dc motor and the other is a stepper motor. The cost of the dc motor is €4 and that of the stepper motor is €0.35. Also, with stepper motor,the control is more accurate. So, the stepper motor with a lead screw is the desired choice of actuator.

3.2.2 Sensors

The actuator chosen does not provide a feedback on its position. Since its a stepper motor, ideally the step count can be traced internally to identify the position of the actuator. But, they have various limiting factors. The step to position ratio of each motor might vary slightly. Also, when the user is touching the pins, due to the pressure from the hands of the user, the pin might not be in the position as intended by the controller. This brings a need for sensors in the system.

Each pin needs its own sensor to track its position. So, to maintain the resolution of the display, the sensor size should be very small, in the order of few millimeters, but measure displacements of a few centimeters as well. This limits the option for the sensor that can be used.

Linear potentiometers

These are displacement sensors which works based on voltage divider and electrical resistance. This is a very economical option and these can vary in different sizes. But, these are contact sensors, meaning the potentiometers will require an electrical medium of connection to the actuator.

Capacitive sensor

These work on the principle of changing capacitance between two plates when they move. They are very compact, but are limited by the actuator stroke length. They are ideal for actuators within 10 mm, and thus not suitable for this display.

Inductive sensor

This works on the change in inductance when there is a movement caused in a flux concentrating element. These are ideal for many operating scenarios as they are available for many different stroke lengths and are non-contact type. But, inductive elements are bulky and as the stroke length increases, they are more susceptible to electro-magnetic interference.

Hall effect sensor

They convert strength of magnetic field to a readable voltage value. This magnetic field is produced when magnets attached to the moving actuator moves during actuation. These are best suitable for short displacement of less than an inch thus making it unsuitable for this application.

Time-of-flight sensors

This works similar to SONAR where it sends out a signal like sound or light and based on the reflection received, it takes the time of flight and calculates the distance. These are ideal for long range applications of a few meters. This is also, not suitable for the chosen application.

Pulse encoding sensors

These are incremental sensors which optically reads the tick marks on a scale to sense the position of the actuator. They are very precise measuring devices and are ideal for 1 to 30 cm of displacement and they are very compact making it a good option for this application.

One such pulse encoder is a photointerrupter. It is a photosensor which has a light emitter on one end and a light receiver on the other end. A photo reflector has both the emitter and receiver on the same end and detects the reflection of the emitted light. This can detect markings on an actuator pin, thus counting the steps taken by the pin.

The ultraminiature photoreflector SPI-315-34 by *SANYO* is one such sensor. Its dimensions are 3.4(L)×2.7(W)×1.5(H) mm making it a really compact design suitable for this application. So, this is chosen as the sensor for this design.

3.3 Prototype design

A pin in the display consists of an actuator, the pin-head and the sensor. This prototype design is a mechanical design that defines how these pins can be

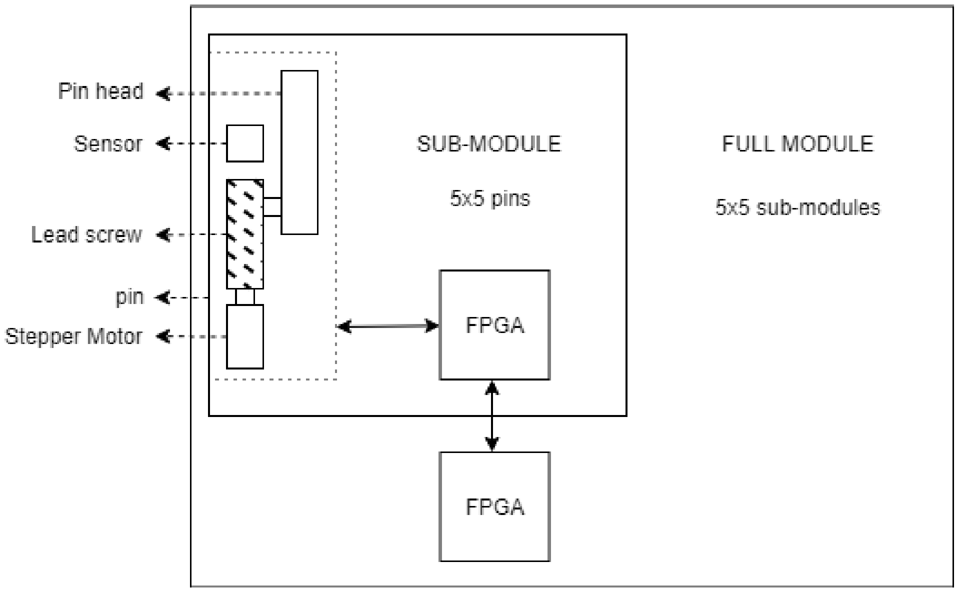


Figure 3.5: **Prototype block diagram**

stacked to make it efficient. There are two models designed. Both of them have a similar sub-module arrangement but differ in the way the actuators are placed.

In both the designs, the whole module is divided into sub-modules. Each sub-module will have a 5×5 grid of pins. The actuators in the sub-module are controlled by one FPGA which will have the actuator control logic. Multiple such sub-modules can be combined to form the full module. The sub-modules are connected and controlled by the primary FPGA which will have the application based logic.

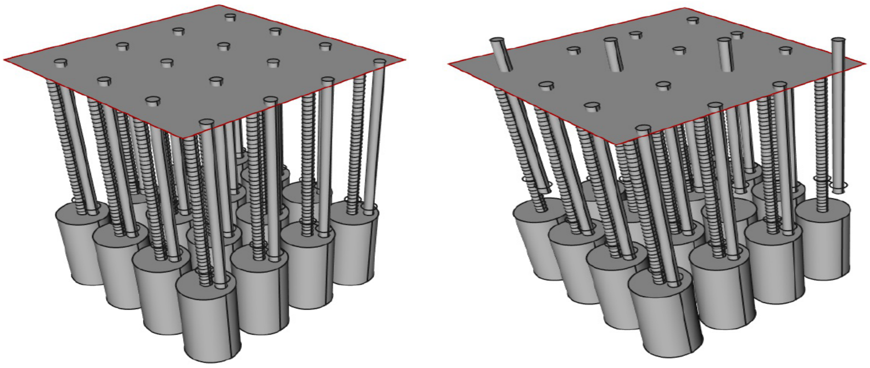
This model is represented in the figure 3.5.

3.3.1 Parallel model

The name parallel model refers to the way the actuators are arranged. They are placed side by side in a horizontal plane to get a 5×5 grid. This is a simple design and can provide a resolution of 5 mm with the chosen actuator. This is simpler to control as the arrangement of pins are not complex. The figure 3.6 shows how the sub-module of this type of model will look like.

3.3.2 Stacked design

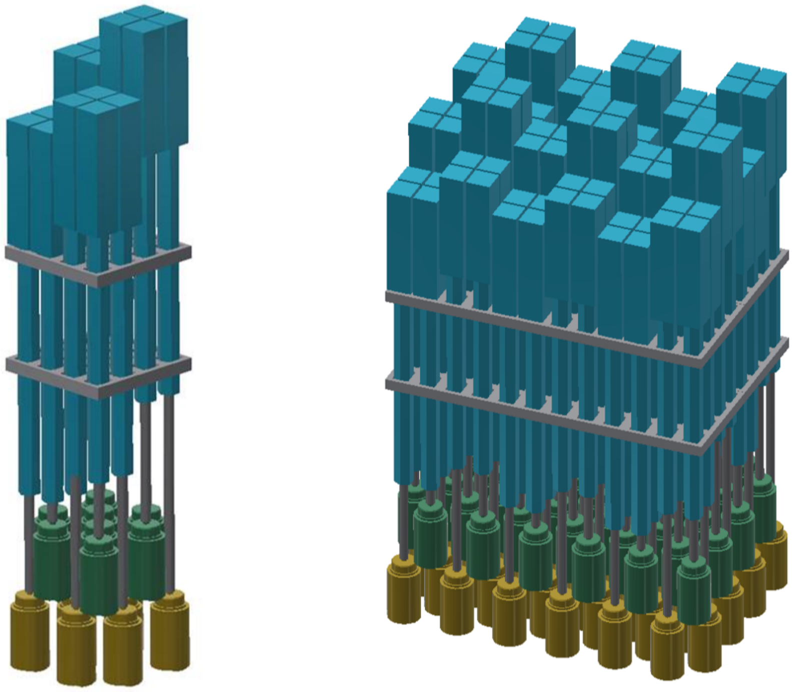
In this model, the actuators are stacked in two rows. In this kind of arrangement, the pins can be placed closer providing a resolution of 3-4 mm. This makes the design more compact, but a bit more complex. The design and manufacturing cost for this model will be higher, but it is a good trade-off if the application requires this resolution. Since the pins are placed in two stacks, the control will also be more complex. The figure 3.7 shows how the model will be. The figure



(a) **All pins at rest** (b) **Pins in movement**

Figure 3.6: **Parallel design**

3.7a shows the sub-module design while the figure 3.7b shows the corresponding full module design.



(a) **Sub-module** (b) **Complete module**

Figure 3.7: **Stacked design**

3.4 Cost estimate

With the help of the *DEMO* team in *Delft University of Technology*, the cost of prototype design was estimated. This helped us get funds approved from

Eindhoven University of Technology. The table 3.2 shows the cost estimate for building a display with 125×125 pins. This does not include the production cost.

Sl. No.	Item/Job	Cost
1	Mechanical design - 1 technician working at TU Delft for 8 weeks	12,800
2	Cost of 1 motor	0.2
3	Cost of sensor	0.5
4	Cost of driver module	0.23
5	Cost of 1 actuator(motor,sensor,driver,gears,pin-head)	~1
6	Cost of PCB for sub-module	~20
7	Battery cost	~3
8	Cost of 1 sub-module	~55
9	Cost of full module	~1400
10	Total cost	**~15,000**

Table 3.2: **Cost estimate**

3.5 Chapter summary

In this chapter, we saw what are the main design requirements and we chose the actuators and sensors to satisfy these requirements. Using these chosen actuators and sensors, we proposed two designs of the 2.5D shape display to get different resolutions. Finally, we provided an estimate of cost to build this display after discussions with the field experts.

Chapter 4

Defining Safety and Comfort

The Cambridge Dictionary defines **Safety** as:

"A state in which or a place where you are safe and not in danger or at risk."[7]

and **Comfort** as:

"A pleasant feeling of being relaxed and free from pain"[4]

The aim of this thesis is to design a safe 2.5D tangible shape display. The word *tangible* means the user can interact with the device via touch. This means that the device has to follow the safety guidelines. But what is safety in a shape display? Do we have any rules or standards that need to be followed? In this chapter, we define some safety guidelines for the chosen hardware by experiment and survey.

Refresh rate is an important parameter for the design of the display. This decides the application of this display in dynamic scenarios. But, with an uncontrolled way of moving the actuators and with increase in refresh rate, the chance of an injury is very high. A single pin moving at the highest speed can cause discomfort and in some cases, tear the skin as well. Because of this, the speed and the way the actuators move needs to be controlled.

Every individual has different levels of sensitivity and tolerance to pain. This varies across gender, age, and even their work background. Some people are just more sensitive to touch by birth. Also, when the sense of touch is helped by other senses such as vision, the user may anticipate the contact before it happens and may be comfortable with higher speed because of this. So, when this varies across individual and environment, the display settings should also be custom-tailored to the user. But, in addition, a safety value should also be imposed.

To understand how this varies among users, a set of people were asked to participate in a survey where they experience the hardware actuation and give a feedback on what they are comfortable with. Two sets of surveys were done

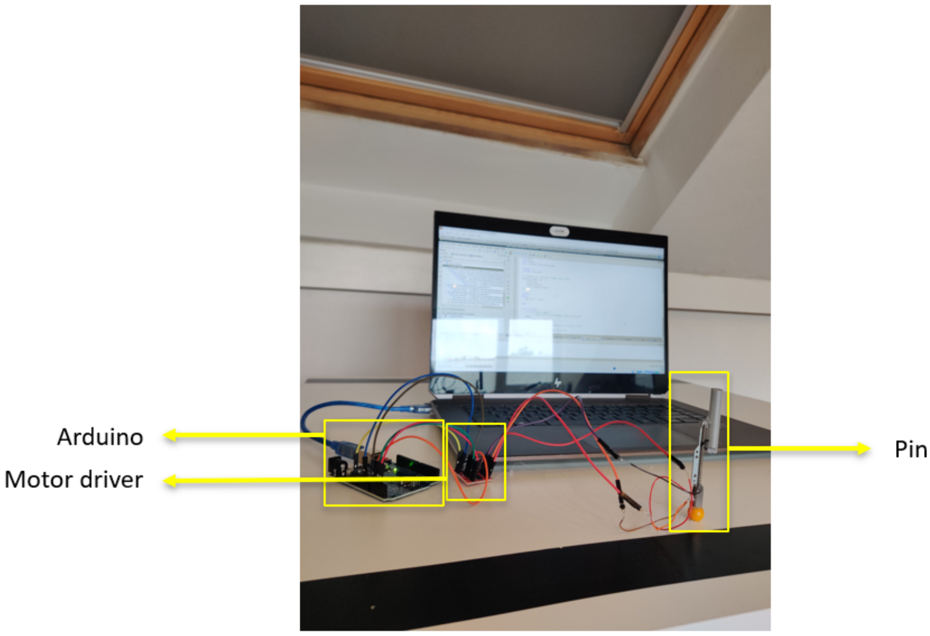


Figure 4.1: **Experimental setup**

to understand this. More the number of people participating in the survey, better will be the result. But, due to Covid-19 situation across the country, the number of participants were limited.

4.1 Experimental setup

The actuators we chose in chapter 3 were used to create a small prototype. A 2×2 pin setup was created with the stepper motor and lead screw coupled to a pin arrangement. The resolution of the prototype was 5 mm. This was controlled by Arduino UNO boards connected via MX1508, a dual H-Bridge motor driver. The setup is shown in figure 4.1.

4.2 Single pin survey

This was a primary survey done to understand sensitivity to the chosen hardware. The goals of this survey are:

1. Understand the maximum speed an individual is comfortable with.

2. Understand how visual perception affects the sensitivity

The second goal was added for development of applications aimed for the visually impaired.

4.2.1 Participants' data

The accuracy of the results of a survey highly depends on the number of participants for the survey. But, what is good number to conclude accurate results? This in itself, is a topic of discussion. According to Sekaran [24], for a good experimental survey, there should be a minimum of 30 participants. But, according to Cohen [11] and Gall [12], the minimum number is 15 for one variable to be evaluated. This increases with the number of variables under consideration.

In this case, the total number of participants was limited to 15 due to the Covid-19 situation. Of the 15 people who participated in the survey, 6 were women and 9 were men. Their ages varied from 21 to 60. They were from different parts of the world, across Asia and Europe. This gave a cultural diversity in the participants.

4.2.2 Experiment

There were two parts to the experiment. In the first part, participants took part with their eyes open and in the second part, with their eyes closed. One pin setup was used in the experiment. The step by step procedure of the experiment is as follows:

1. The participant was asked to keep their hands at the top of the pin.
2. The actuator was moved with a random speed between 1 mm/s to 8 mm/s.
3. The participant was asked if they were comfortable with the actuation and if they felt safe to keep their hands there.
4. This was repeated for about 10 times to understand what range of speeds the participant was uncomfortable with.
5. If the participant was comfortable with 10 mm/s as well, then higher speeds were used.
6. Once the range of discomfort was found, a speed range including a few comfortable range and a few uncomfortable range was chosen. For example, if the participant was uncomfortable at 6-8 mm/s in the initial test, the speed range chosen will be 4-10 mm/s. This will give the participant a range of values to test against. Also, with a bigger range, noticeable change in speeds can be applied to accurately evaluate the comfort.
7. Next, the initial experiment was repeated multiple times with speed given to the user within this range.
8. In between each actuation, the user was asked to remove the hand so that the reason for comfort will not be a memory from previous actuation.
9. Once the participant was uncomfortable with a particular speed at least twice more than the previous speed value, this was chosen as the value of discomfort.
10. This experiment was then repeated with closed eyes of the participant.

4.2.3 Survey results

The speeds the participants were comfortable with for one actuator setup are as outlined in table 4.1.

Participant	Speed Tolerance (mm/s)	
	With eyes open	With eyes closed
P1	12	8
P2	13	12
P3	12	10
P4	9	8
P5	12	9
P6	11	9
P7	10	10
P8	12	8
P9	10	9
P10	13	9
P11	11	10
P12	8	9
P13	12	10
P14	9	7
P15	13	11

Table 4.1: **Single pin survey results**

The results are also visualized in the histogram plot in figure 4.2.

From the histogram plot, we can see an evident shift in the plot when the person cannot see the actual pin in motion. The plot also shows that when the participant was able to see, the majority of them were comfortable with speed of 12 mm/s. When the eyes of the participant was closed, the result seems to vary from 8 mm/s to 12 mm/s. This sudden decrease in tolerance and uncertainty could be because of the anticipation and the fear that it might not be comfortable or safe.

Along with the experiment data, the participants were also asked what could make them feel more comfortable and safe to use the display even if they were visually inaccessible. Some of the feedback were:

1. The size of the pin being too small was major concern for most of them. They were afraid if it might cause damage to their skin.

2. An accelerating speed might help as they will be able to anticipate the movement of the pin once it starts to move.

3. Sudden and high acceleration was uncomfortable.

Based on the feedback and the results of the survey, five different safe operating algorithms for the display were formulated. This will be covered in later chapters.

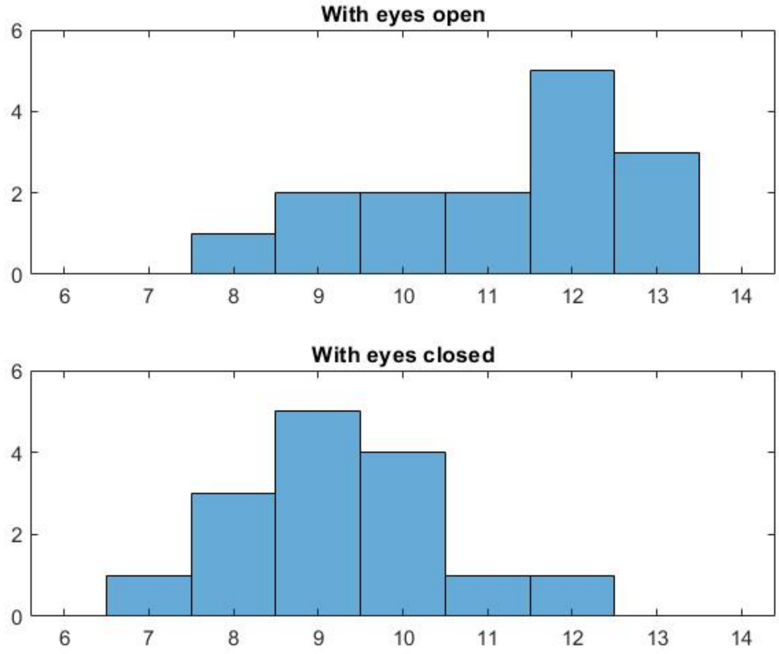


Figure 4.2: **Single pin survey result - Plot**

4.3 Multi-pin survey

Based on the result and feedback from the single pin survey, another experiment was conducted to understand the effect of size and speed of the actuator on the comfort of the user. The goals of this experiment and survey are:

1. To understand how the size of the actuator affects the comfort of the user
2. Get a quantitative relation to size and speed of actuator, and comfort and safety of the user

4.3.1 Participants' data

A total of 15 participants, same as the previous survey took part in this survey as well. This number was also limited due to Covid-19 limitations. The group comprised of 8 women and 7 men. Their ages varied from 21 to 60. They were mostly from Europe and Asia. This survey can also help understand the sensitivity among people in different age group.

4.3.2 Experiment

In this experiment, the setup similar to single pin survey set up was used. But, instead of single actuator, 4 actuators were used. The same experiment as the single pin survey was repeated. The participant started with a single actuator test to get a comfortable speed for the participant. This was later repeated with two, three and four pins moving synchronously.

The participants were asked their feedback to determine the line of comfort and discomfort. This value was noted to create a safety and comfort guideline for the system.

4.3.3 Survey results

The table 4.2 shows the result of the multi-pin survey. This is also visualized in the histogram plot in figure 4.3.

Participant	Speed Tolerance (mm/s)			
	1 pin	2 pins	3 pins	4 pins
P1	9	11	15	15
P2	12	15	15	15
P3	7	9	10	12
P4	10	12	14	15
P5	9	10	12	14
P6	12	14	14	15
P7	8	11	14	15
P8	11	12	12	14
P9	11	13	15	15
P10	13	13	15	15
P11	10	13	14	15
P12	10	12	13	13
P13	12	12	14	14
P14	11	14	15	15
P15	12	13	15	15

Table 4.2: **Multi-pin survey results**

From the plot, we can see that the histogram shifts to higher speed as the size of the pins, that is, the number of pins increases. The following conclusions can be drawn from the survey results:

1. The participants are comfortable with higher speeds as area increases.
2. From the previous point, we can understand that the comfort of the user depends on the pressure exerted by the actuator.
3. Most of the users are comfortable with the highest speed of 15 mm/s when 4 pins are moved synchronously. This means that, we can safely say that as the number of pins moving synchronously increases, any user we consider should be comfortable with the actuation.

4.4 Chapter summary

In this chapter, we saw two different experimental surveys conducted to understand what can be a good safety and comfort guideline for the prototype we designed in chapter 3. These results will be later used in chapter 6 to find the best methods of actuation for the entire system.

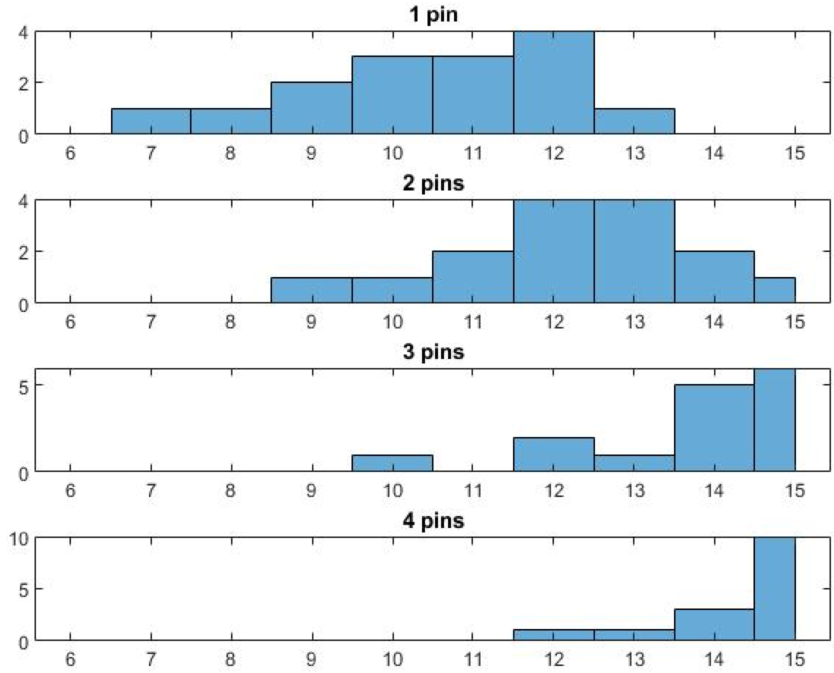


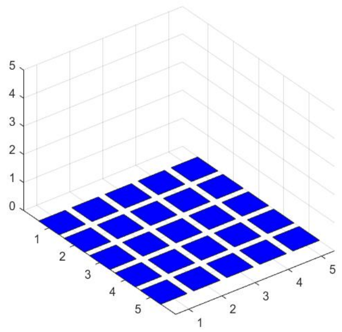
Figure 4.3: **Multi-pin survey result - Plot**

Chapter 5

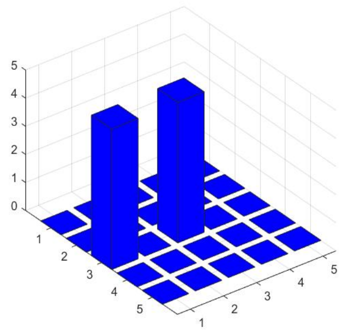
Safe Actuation Algorithms

This chapter outlines the algorithms that we develop for the safe actuation of the 2.5D display designed in chapter 3. These will be implemented on the sub-module FPGA in the design. The FPGA will receive the final image state and the current image state as input and based on the algorithm loaded, it will generate control signals for the operation of the motors.

Based on the single pin survey conducted in chapter 4, section 4.2, we design five different algorithms. These algorithms decide how the pins move, that is, their speed and pattern to change. For example, say the display needs to create image 5.1b from 5.1a, then there are multiple ways this can be achieved. For example, it can follow the pattern as in figures 5.2, 5.3 or 5.4. Each of these create the final image but vary in the way the image is created.



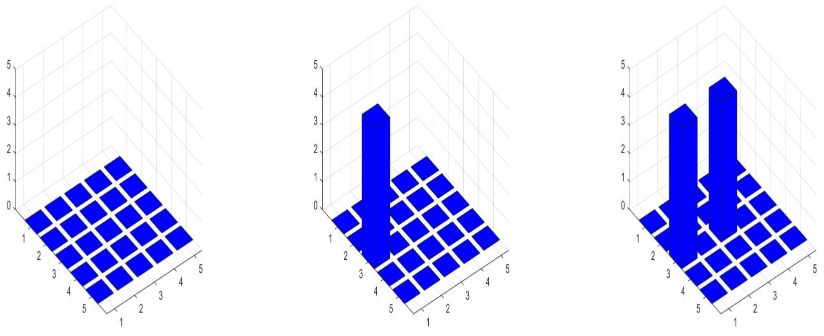
(a) **Image A**



(b) **Image B**

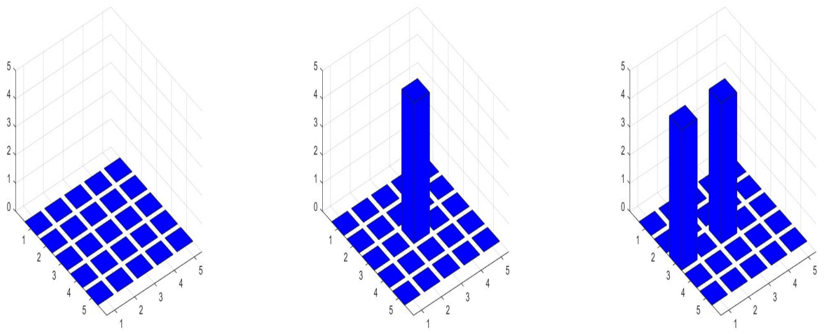
Figure 5.1: **Example images**

We initially discuss The fastest algorithm which is used as a baseline to compare the advantages and disadvantages of the other algorithms that will be discussed. The following sections explain each algorithm, including the basic algorithm with an example.



(a) **Image A** (b) **Image B** (c) **Image C**

Figure 5.2: **Example algorithm 1**



(a) **Image A** (b) **Image B** (c) **Image C**

Figure 5.3: **Example algorithm 2**

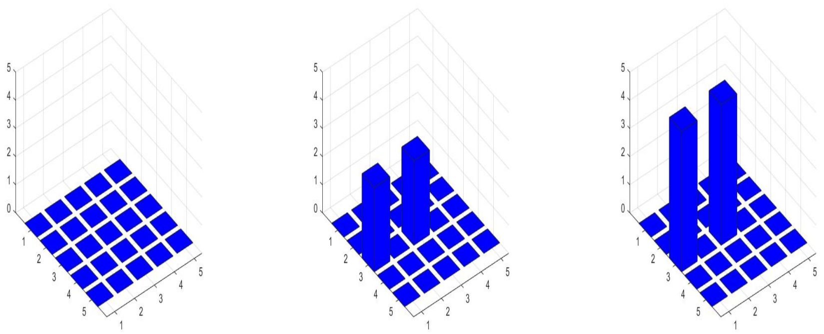
5.1 The fastest algorithm

This is the basic algorithm against which we will compared all the other algorithms. This works on a very simple logic.

5.1.1 Working

In this algorithm, to move from an Image A to Image B, all the actuators not in the final position, will start to move at the same time at the maximum speed towards the final position. The pins moving upwards and downwards will move at the same speed of 15 mm/s.

Based on the results of the multi-pin survey, conducted separately for the user, a customizable safety cap for the maximum speed has been added as a variable. This means that the speed will then drop to the cap set by this variable.

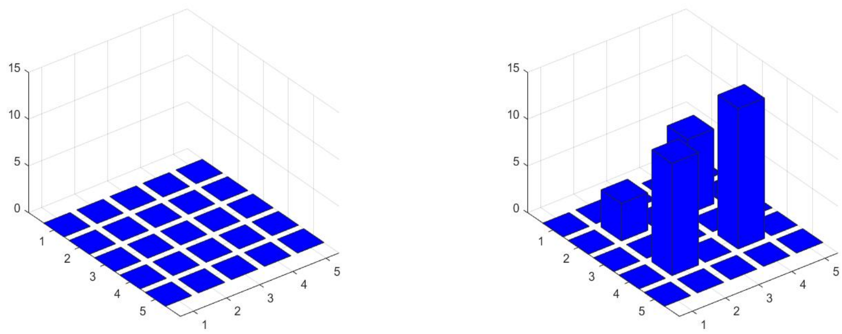


(a) **Image A** (b) **Image B** (c) **Image C**

Figure 5.4: **Example algorithm 3**

5.1.2 Example

In this example the timeline for the movement of actuators from Image A to Image B in figure 5.5 using the fastest algorithm is explained.



(a) **Image A** (b) **Image B**

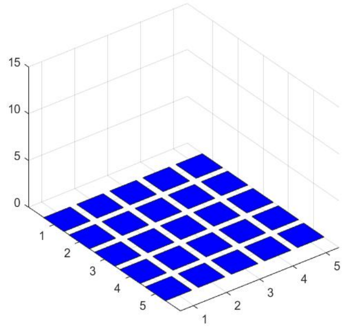
Figure 5.5: **Initial and Final Images**

The figure 5.6 shows the way the actuators move in time according to the fastest algorithm for the example. Each step is 1.5 mm and 1 unit of time is 0.1s. From the figure, we can see that the pins are moving at a constant speed. They stop when they reach the final position.

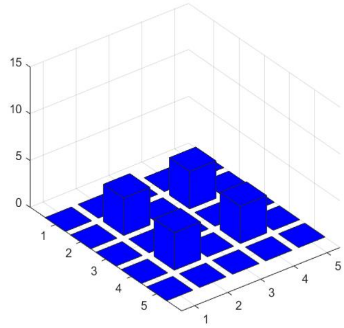
5.1.3 Discussions on the fastest algorithm

We saw in the previous section that this algorithm works at the maximum speed for all pins from time zero, making this the fastest algorithm. In this algorithm, since only the pins that are required to move will be actuated, this also has the least power consumption.

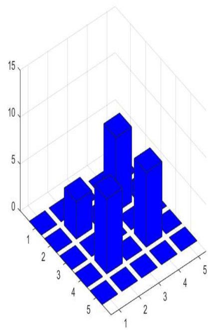
In an ideal case where safety and comfort is not of concern, this will be the best algorithm in terms of time and power. But, with the display having a



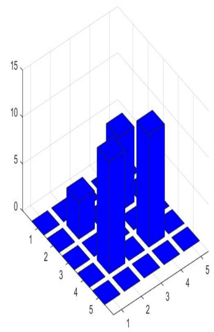
(a) **t=0s**



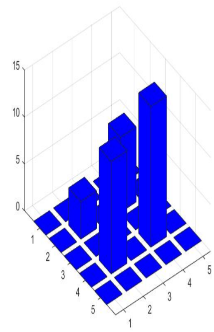
(b) **t=0.4s**



(c) **t=0.8s**



(d) **t=1.2s**



(e) **t=1.5s**

Figure 5.6: **The fastest algorithm working**

tangible benefit, there is a need to design other algorithms with safety being a higher priority.

5.2 Accelerating algorithm

At the end of the single pin survey, some of the participant's feedback mentions how they will be comfortable if the pins moved slowly at first, so they can anticipate the pin movement before the speed increases. Based on this feedback, we designed the accelerating algorithm.

Unlike the fastest algorithm, this algorithm is designed such that the user will be able to anticipate an actuation and prepare for the movement. This will be able to reduce the risk in comparison.

5.2.1 Working

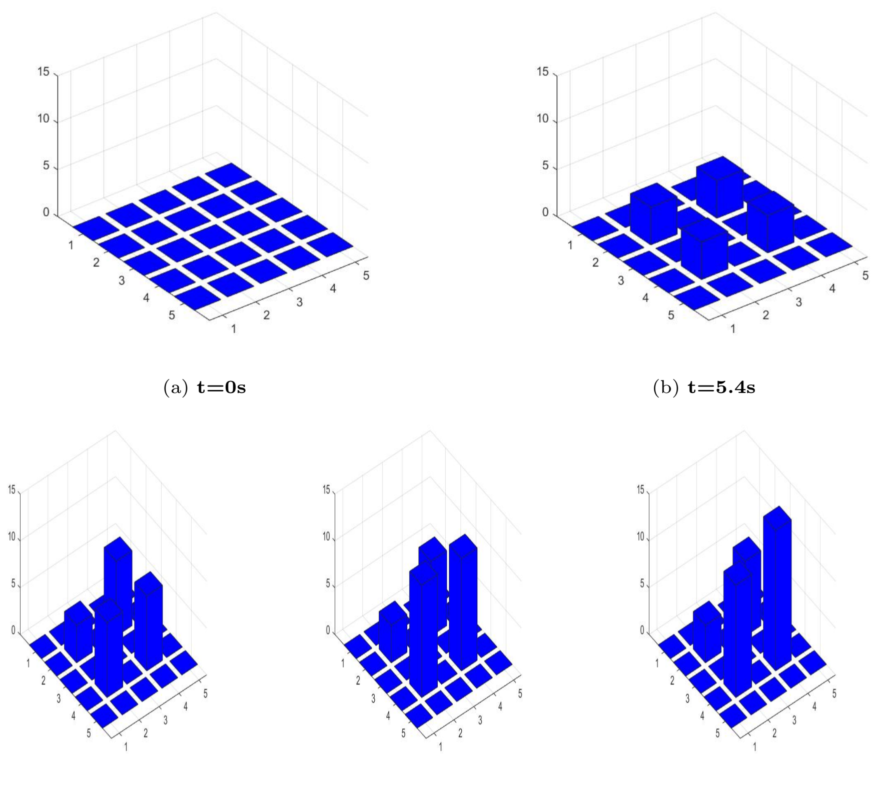
In this algorithm, there are 15 possible speeds for the actuator ranging from 1 mm/s to 15 mm/s. In this, the actuators initially starts moving at the minimum speed of 1 mm/s and the speed increases in uniform steps to the maximum speed of 15 mm/s.

The algorithm is designed such that a safety cap speed limit can be added to suit the user's comfort. This will limit the maximum speed of the display to the safety value. When the actuators reach this speed, they continue to maintain the speed until they reach the final position.

5.2.2 Example

The example from the fastest algorithm in section 5.1.2 is taken with the initial and final images shown in figure 5.5. The section explains the timeline with the accelerating algorithm control of the actuator.

Figure 5.7 shows the position of the actuator at different points of time. From the image, comparing it to the fastest algorithm shown in figure 5.6, we can see that the actuators are accelerating as they move up. The total time taken to transform the image has also increased by 8 times. This shows how there is a trade-off in time when the speed is reduced for the user's comfort and safety.



(a) **t=0s** (b) **t=5.4s**

(c) **t=9.2s** (d) **t=11.4s** (e) **t=12s**

Figure 5.7: **Accelerating algorithm working**

5.2.3 Discussions on the accelerating algorithm

From the previous section example, we can see that this algorithm has a significant increase in time. This is expected with reduction in speed of actuation.

In terms of power efficiency, this will perform almost as same as the fastest algorithm. The need for acceleration in between can create a higher power requirement, but is not a significant increase.

We designed this algorithm with the assumption that the user's hands will always be in contact with the actuator at the start of actuation. But what happen if the user waits for the actuation to be almost over before placing their hands? Then, the logic behind building this algorithm is lost. This leads us to the reason for the design of the next algorithm, the decelerating algorithm with the opposite logic as the accelerating algorithm.

5.3 Decelerating algorithm

This is the opposite of the accelerating algorithm. This was designed to understand the effect of decelerating displacement in the display.

In the previous session, we designed the accelerating algorithm since users were comfortable with acceleration when hands were placed at the top of the actuators. Now, what happens when the hands are placed higher away than the actuators? Then, wouldn't it be better to have the actuators slow down when it nears the hand position? This algorithm was designed with this particular scenario in mind.

5.3.1 Working

Similar to the accelerating algorithm, there are 15 different operating speeds for this algorithm, varying from 1 mm/s to 15 mm/s. In this, the actuator starts to move at its maximum speed of 15 mm/s and reduces its speed uniformly in steps to the minimum value of 1 mm/s.

There is a maximum value of speed that can be added to cap the starting speed to this value. This will depend on the individual user and his/her level of comfort.

5.3.2 Example

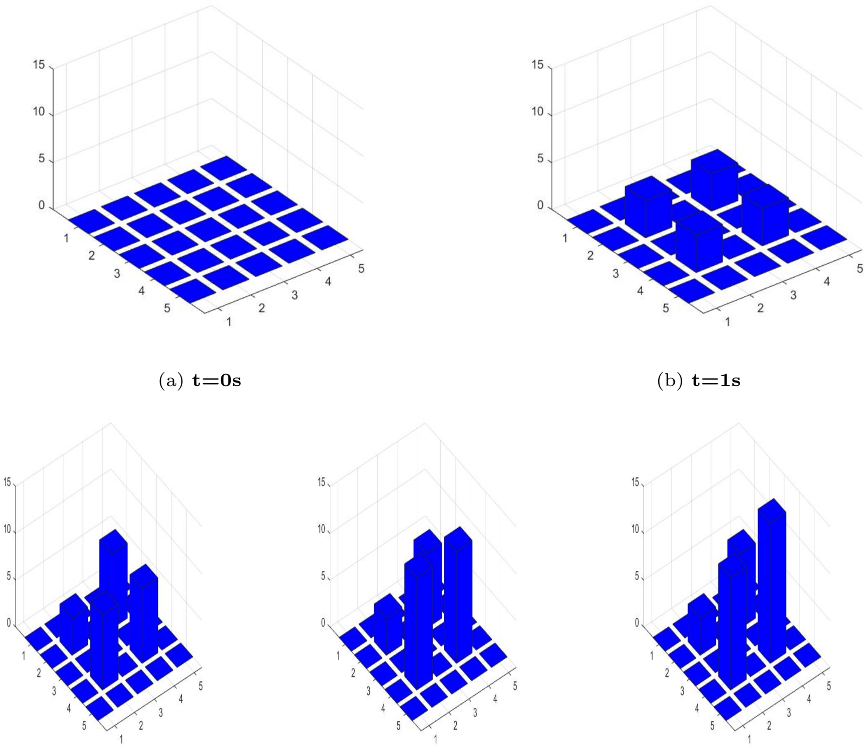
The same example from the fastest algorithm in section 5.1.2 is chosen with the same starting and final image as in figure 5.5.

The timeline plot of the algorithm for this example is shown in figure 5.8. We can see that the total time is same as the accelerating algorithm, but the initial states are reached faster. This shows that the actuators are moving fast initially and decelerates with time. Time-wise, this performs similar to the accelerating algorithm.

5.3.3 Discussions on the decelerating algorithm

This algorithm as seen from previous section will have a similar time and power efficiency as the accelerating algorithm. The power might vary slightly due to the effect of deceleration, but the value is insignificant.

This algorithm might be effective when user has his hands hovering over the actuator. But, what will be the effect if the hands are placed over the actuator. Then, the actuators will have a high acceleration due to the high speed creating



(a) **t=0s** (b) **t=1s**

(c) **t=3.6s** (d) **t=7.8s** (e) **t=12s**

Figure 5.8: **Decelerating algorithm working**

an enormous force. This will be very unsafe for the user and can even lead to injuries. So, the use of this algorithm will have to be very restrictive based on the applications.

5.4 Sea wave algorithm

The results of the multi-pin survey shows that when the number of pins moving synchronously are more, the user can tolerate higher speeds. Based on this result, the sea wave algorithm was designed to give maximum comfort to the user, while providing a balance in time-efficiency as well. The name *sea wave algorithm* is derived from the way it operates.

This algorithm was designed because in all the previous algorithms, we had single actuators moving at highest speed at some point. This is the worst case scenario for the display in regards to safety. To avoid this particular scenario, the sea wave algorithm was built.

5.4.1 Working

Since the users are comfortable with the maximum speed when multiple pins are moving together synchronously, in this algorithm, initially we move all the pins in the display to the maximum height in final image. Once all the pins reach the highest required point, the pins start moving to their final positions. All these actuations are done at the maximum speed of 15 mm/s.

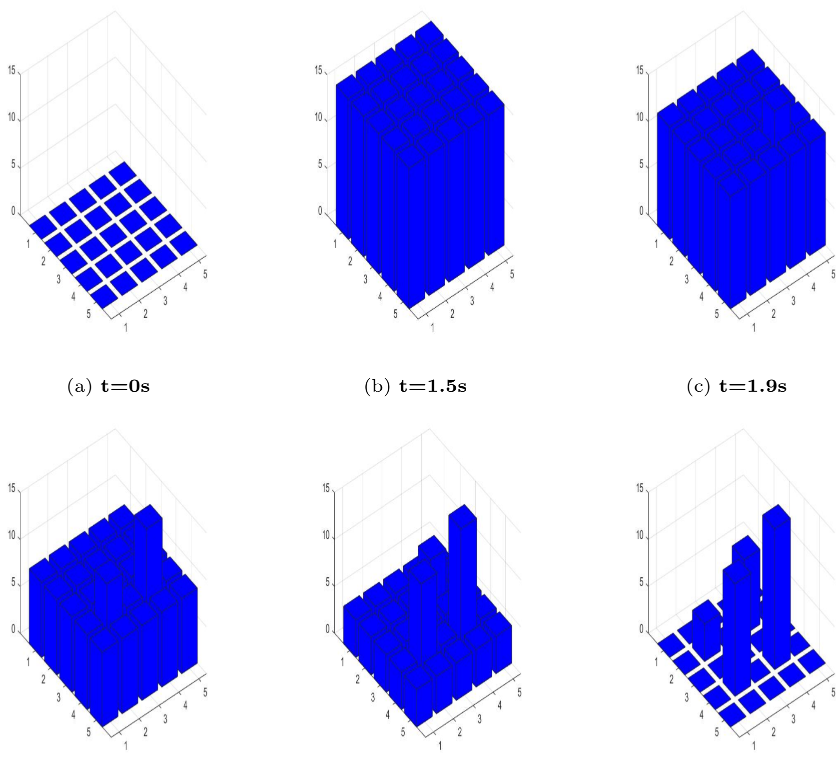
The safety limit of a maximum speed value is not required here since the survey shows the maximum speed is comfortable with more than 4 pins for most of the participants. This can be extrapolated to understand that when a big display of more than hundred pins are moving synchronously, the maximum speed should be beyond comfortable.

5.4.2 Example

The same example from section 5.1.2 is used for this algorithm. The initial and final positions are as shown in figure 5.5.

Figure 5.9 shows the timeline plot for the chosen example with the sea wave algorithm as the control algorithm. Figure 5.9b shows all the pins reaching the highest point and figures 5.9c, 5.9d and 5.9e show how the pins fall back leaving the the pins in their respective final positions.

Figure 5.9 also shows that the total time for the execution of this example is 3.1 s, which is nearly 4 times faster than the accelerating and decelerating algorithms, and about twice the time taken by the fastest algorithm.



(a) **t=0s** (b) **t=1.5s** (c) **t=1.9s**

(d) **t=2.3s** (e) **t=2.7s** (f) **t=3.1s**

Figure 5.9: **Sea wave algorithm working**

5.4.3 Discussions on the sea wave algorithm

In terms of time-efficiency wise, this algorithm performs well. But it is inefficient in terms of power usage. This is because even when one pin needs to move up, hundreds of motors will be operating synchronously. This is a huge waste of energy. For any image scenario we consider, this will have a worse power consumption than the previously discussed algorithms.

How can we decrease the power usage while still having the same standards of safety? Fr this, the number of motors operating simultaneously will have to reduce. With this objective, we design the segmented sea wave algorithm.

5.5 Segmented sea wave algorithm

This is a truncated version of the sea wave algorithm. The aim of this algorithm design is to provide similar functionality and advantages as the sea wave algorithm, while reducing the total power consumption.

Most of the applications in today's world are aimed at reducing power consumption. Also, if we need to make the device portable, efficient usage of battery power is very important. This algorithm tries to provide better power efficiency with better safety as well.

5.5.1 Working

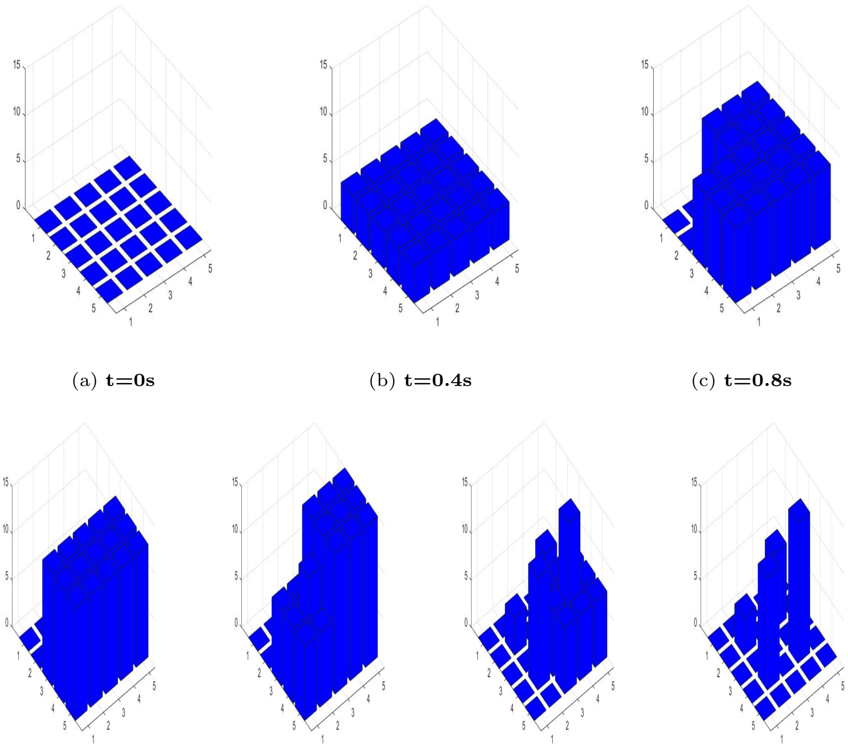
We saw that moving all the pins to the highest position was the major cause of energy inefficiency in the sea wave algorithm. So, in this algorithm, this part is replaced. Instead of moving all the pins to the highest position as in sea wave algorithm, the pins surrounding a moving pin will move along with it to final position of the center pin. Then, the surrounding pins will drop to its own final position. The speed at which these pins move will remain the same at 15 mm/s.

5.5.2 Example

The same example from section 5.1.2 is chosen for this algorithm as well. The initial and final images remain the same as figure 5.5.

Figure 5.10 shows the working of the segmented sea wave algorithm for this chosen example. Figure 5.10b shows the pins reaching the final position of one of the pins at t=0.4s. Figure 5.10c shows the pins adjacent to the first pin has moved back to the ground plane leaving the pin in its position. The other pins continue to move to the final position of the second pin. Figures 5.10d, 5.10e and 5.10f shows the same working for the other 3 pins.

In this example, we can see that the final time remains the same as that of the sea wave algorithm while operating lesser motors for the segmented sea wave algorithm. In this algorithm as well, at least 9 pins move synchronously upwards to create the final image. From the multi pin survey, it can be extrapolated that the user should still be very comfortable with 9 pins moving at the highest speed of 15 mm/s.



(a) **t=0s** (b) **t=0.4s** (c) **t=0.8s**

(d) **t=1.2s** (e) **t=1.6s** (f) **t=2.4s** (g) **t=3.1s**

Figure 5.10: **Segmented sea wave algorithm working**

5.5.3 Discussions on the segmented sea wave algorithm

From the discussion of the example in previous section, we can see that this algorithm provides same time efficiency as the sea wave algorithm. But the power efficiency is reduced significantly. In a 5×5 display of pins, if one pin needs to move up, in the sea wave algorithm, all the 25 pins will move up. But in the segmented sea wave algorithm, only 9 pins will move up. This has reduced the power usage by almost two-thirds.

But, can safety be still increased with increases power efficiency? For this purpose, we look back to the multi pin survey results. We can see a trend in the histogram plot in figure 4.3. We see that as the area of the pins increase, the participant is able to withstand higher speeds. The next algorithm is designed with this as the basis.

5.6 Speed vs Area algorithm

The multi pin survey shows that participants were comfortable with higher speeds as the number of pin operating synchronously increased. This was the basis of designing this algorithm.

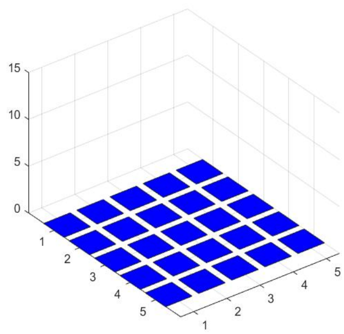
The need for this algorithm rises to create a safe algorithm while not compromising on the power efficiency of the system.

5.6.1 Working

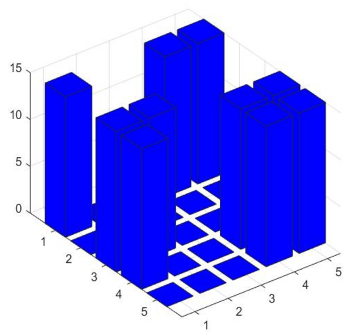
In this algorithm, the speed of the actuator will be calculated based on the number of adjacent actuators that will be moving along with it. If all 8 of the surrounding pins were moving, then it will move at the highest speed of 15 mm/s. As the number of adjacent pins moving reduces, the speed also reduces uniformly in steps.

5.6.2 Example

A different example is chosen to clearly understand the working of this algorithm. The initial and final images shown in figure 5.11 are chosen for this example. It can be seen that in this example, there are sets of 1 pin, 2 pins, 3 pins and 4 pins that needs to be moved to their final positions.



(a) **Image A**



(b) **Image B**

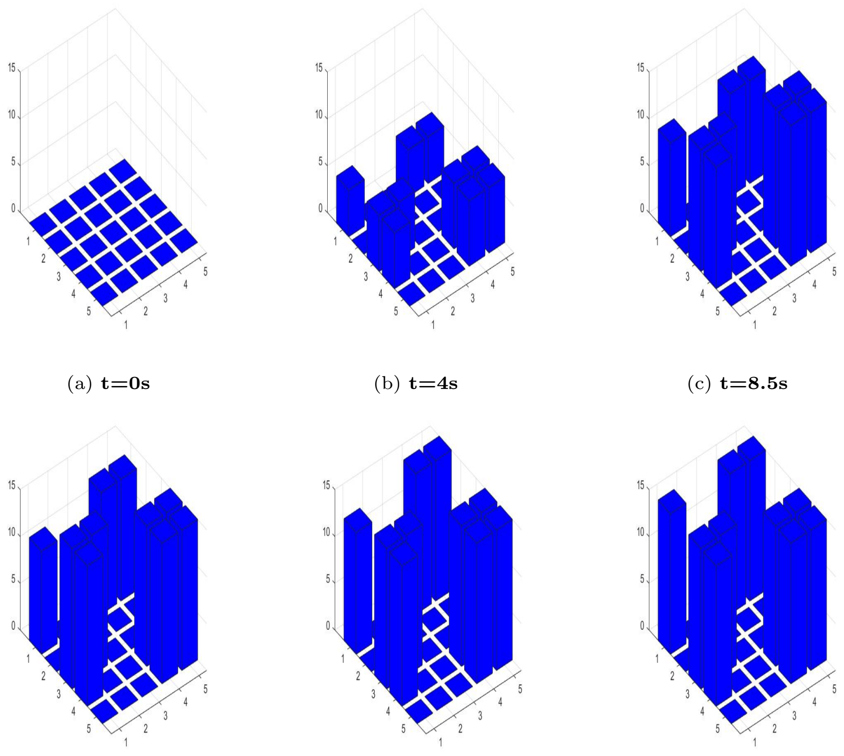
Figure 5.11: **Initial and final images**

Figure 5.12 shows the timeline plot for this example created using the speed vs area algorithm. The figure 5.12b shows the pins at different heights due to the different speeds at which they are moving. Figures 5.12c, 5.12d, 5.12e and 5.12f shows the pin reaching their final positions at times 8.5 s, 9.9 s, 11.2 s and 13.5 s.

5.6.3 Discussions on speed vs area algorithm

The total time taken for the execution using this algorithm is the highest among all the other algorithms. But this is image dependant. This high time was caused due to the single pin moving separately. This time efficiency will be better if based on the application instance.

Even with a higher time, this might be a better algorithm for most of the applications to reduce the power while maintaining the safety and comfort. The time is a trade-off that needs to be accepted if necessary for the application.



(a) **t=0s** (b) **t=4s** (c) **t=8.5s**

(d) **t=9.9s** (e) **t=11.2s** (f) **t=13.5s**

Figure 5.12: **Speed Vs Area algorithm working**

5.7 Chapter summary

This chapter covered all the algorithms designed with the aim of increasing the safety and comfort of the user. In all the cases, there was a trade-off required in time, thus affecting the refresh rate of the display.

The fastest algorithm will have the best refresh rate, but the least safety. The sea wave and segmented sea wave algorithms have the next best refresh rates, but the segmented sea wave algorithm performs better in terms of power efficiency. The accelerating and the decelerating algorithms has a refresh rate which is about four times lesser than the sea wave and segmented sea wave algorithms. The speed vs area algorithm seems to have the worst refresh rate, but this is application dependent. The refresh rate would have been much higher if there was no single pin moving separately. So, this can perform similar to the other four algorithms depending on the application scenario.

Some more advantages and disadvantages of these algorithms are analysed in chapter 6. Chapter 6 will explain how each algorithm is evaluated and will be analyzed for different scenarios. Then, it will provide different application scenarios based suggestions on what can be the best algorithm in terms of safety and comfort of the users.

Chapter 6

Simulation and evaluation of algorithms

In chapter 3, we had designed a 2.5D shape display, analysed the safety and comfort with an experimental survey on a prototype in chapter 4 and designed six algorithms for safe and comfortable operation of this display in chapter 5. But, how can we say these actuation algorithms are really safe? The best way to validate will be to implement them in the hardware and test with different users. This was the plan for the thesis. We had some funding agreements with Eindhoven University of Technology for building the hardware. But, due to the unforeseen circumstances of the Covid-19, the labs had to be closed and building of the hardware had to be postponed.

So, without an actual hardware, how can we test for safety? The survey done in chapter 4 gave an insight into what are safe operating conditions. But, that was done with a maximum of four pins. We need a way to extrapolate the results of the survey to understand the safety of the full design. For this purpose, we build a simulator that can mimic the mechanics of the display. This will be explained in section 6.1. But even having a simulator, how can we create a metric to understand the safety and comfort in using the designed 2.5D display? This metric has been formulated in section 6.2.2 of this chapter. With this metric, the different actuation algorithms are analysed in section 6.3.

6.1 Simulator Design

The figure 6.1 shows the parts of the simulator and the control logic. It can be seen that the entire working of the 5×5 pins in the sub-module will be simulated and the control logic code will be part of the sub-module FPGA. The pin includes the stepper motor and the sensor, whose signals will be simulated in the simulator.

Figure 6.2 shows the flow of data from simulator to the control logic. It can be seen that, the initial image data, which is the initial position of pins is given as input to the simulator. This represents the initial sensor output. The final image data, that is, the final pin positions are given as input to the control logic FPGA. In the actual setup, this data will be provided by the primary FPGA.

Based on the initial image data received, the simulator will provide the initial

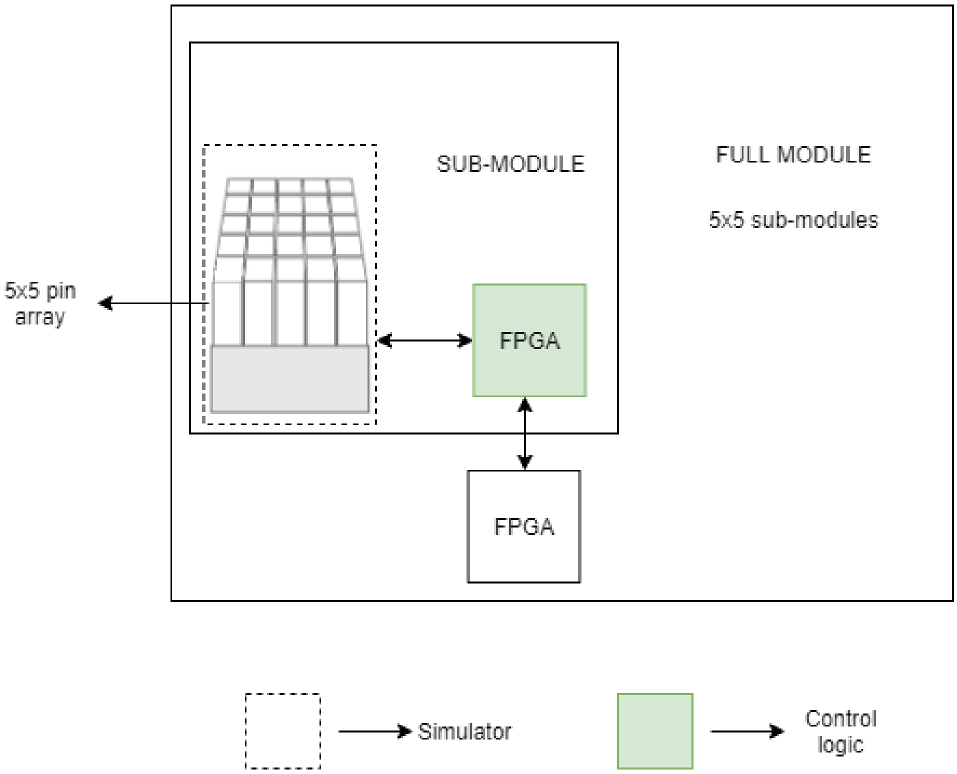


Figure 6.1: **Simulator components**

sensor data to the FPGA. On receiving this data, the FPGA generates the control signal for the 5×5 motor setup based on the algorithm used. With this data, the simulator updates the sensor position info and sends it to the FPGA. This loop goes on till the pin positions reach the final image position provided to the FPGA.

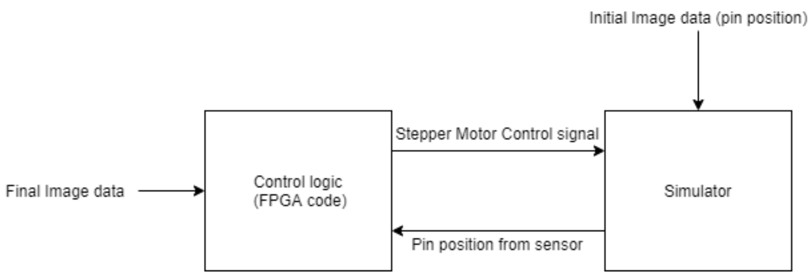


Figure 6.2: **Simulator Data flow**

For the purpose of simulation, the software tools that we use are Vivado simulator and test bench. The FPGA control logic code, which is the logic implementation of the algorithms is implemented on Vivado Design Suite and the working of the stepper motor and sensor are implemented as part of the test bench. The signal data from Vivado is later analysed using LabVIEW and

MATLAB.

The FPGA code hence done is such that it can be directly ported to the hardware.

Now that we have the simulator and FPGA data for the different algorithms to create an image, we can evaluate how this performs.

6.2 Method of evaluation

To evaluate the performance of the actuation algorithms in terms of safety and comfort, we need a metric that can be used to say which is better suited for the given scenario. For this purpose, we use **pressure** exerted by the pins to provide a safety and comfort guideline.

6.2.1 Pressure calculation

For calculating the pressure exerted by the pin when moving, we look into the basics of physics. Figure 6.3 shows the force diagram of the display.

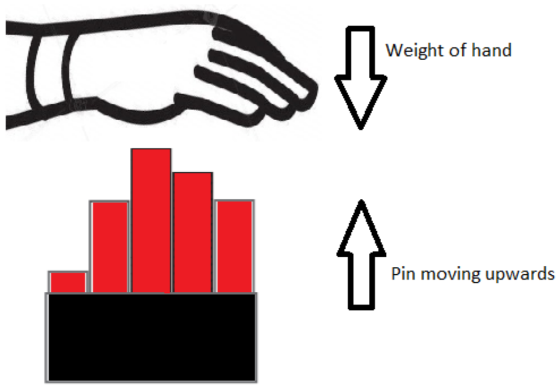


Figure 6.3: **Pressure calculation**

The main forces that contribute to the total force generated are the weight of the hand acting downwards and the force of the pin moving upwards. We use this to derive the equation of pressure exerted as shown.

$$\text{Total Force} = \text{Force exerted by pin} - \text{Force exerted by hand} \tag{6.1}$$

$$\text{Force exerted by hand} = \text{Hand weight} \times g \tag{6.2}$$

where g is gravity.

$$\text{Force exerted by pin} = \text{mass of pin} \times \text{Acceleration of pin} \tag{6.3}$$

$$\text{Pressure} = \frac{Force}{Area} \tag{6.4}$$

$$\text{Total pressure} = \frac{(\text{Hand mass on pin} \times g) - (\text{mass of pin} \times \text{Pin acceleration})}{\text{Pin area}} \tag{6.5}$$

Equation 6.5 is used to calculate the pressure exerted by the pins on the hands of the user. First, we combine this equation with the results obtained from the multi-pin survey to understand how the pressure metric will look like.

6.2.2 Pressure calculation for Multi-pin survey results

We obtained a set speed tolerance values for multi-pin setup from the participants of the survey. Using these values in equation 6.5, we generate the histogram plot shown in figure 6.4. From this histogram, we can see that with 1 pin, pressure tolerance is a bit lesser than with 4 pins. We can have this as the reference to understand the pressure data from simulation.

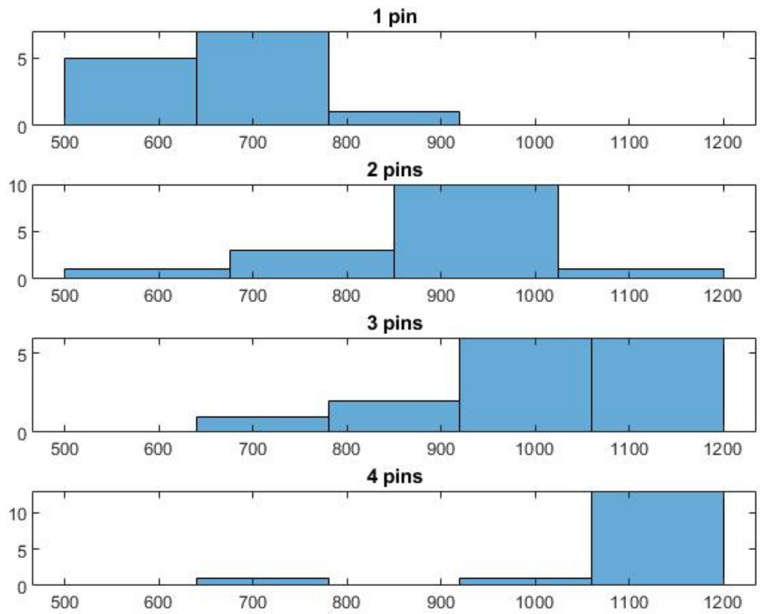


Figure 6.4: **Pressure data from Multi-pin survey**

This is the basic metric that we use to evaluate our actuation algorithms. But, how are we extending the results of survey done on the 4 pin set-up to evaluate the safety and comfort when the whole display is in use? For this, we draw some conclusions from the metric which will be used to evaluate the actuation algorithms. The conclusions that can be drawn from the metric are:

1. The pressure the user is comfortable with increases with increase in the number of pins moving synchronously.

2. With a single pin movement, the users are comfortable with pressures in the range of 500 to 800 kPa. So, pressures below these values are preferred with single pin actuation.

3. For four pin actuation, comfortable pressure range are from 800 kPa to 1100 kPa. Also, the majority of users are comfortable with the highest pressure.

4. Combining points 1 and 3, as we increase the number of pins moving synchronously beyond 4, we can safely assume that, users will be comfortable with higher pressures.

The analysis of the various actuation algorithms based on the derived conclusions are done in the next section.

6.3 Evaluation of actuation algorithms

With the implementation of all six algorithms in FPGA, we can get the stepper motor control signal data to create an image. To understand how these algorithms perform in terms of speed, we look into different application scenarios. Most of the applications can be broadly divided into two categories, namely the static image creation and dynamic image creation.

Static image means we go from one image to another and stay in this state. Dynamic image means the image keeps changing with time like showing how sea waves look like or how an object is moved from one place to another. So, in this section, we look into these two broad scenarios to identify which algorithms can be used for each type of application.

6.3.1 Static image applications

There are many applications for 2.5D shape displays where static images are used. Some such applications are city-scape modelling, physical object mimicking and braille displays. In all these cases, we need a transformation from say image A to image B, but the way it is transformed is not of importance. The user is only interested in the final image.

In chapter 5, we saw an example for each algorithm that shows a static image creation scenario. We will use the same examples to understand the advantages and disadvantages of each algorithm in the case of static image creation.

Fastest algorithm

In section 5.1.2 we saw how the algorithm works for moving from Image A to Image B as shown in figure 5.5. The total time taken to create the image was 1.5 s, which was the fastest among all the algorithms. This is the best in terms of time efficiency.

Figure 6.5 shows the pressure distribution when all the four actuators are moving up. From this figure, we can see that this produces a pressure of more than 1000kPa on one pin. Comparing this with the survey data for 1 pin, we see that this is a really high pressure. So, this is an unsafe algorithm for static image creation.

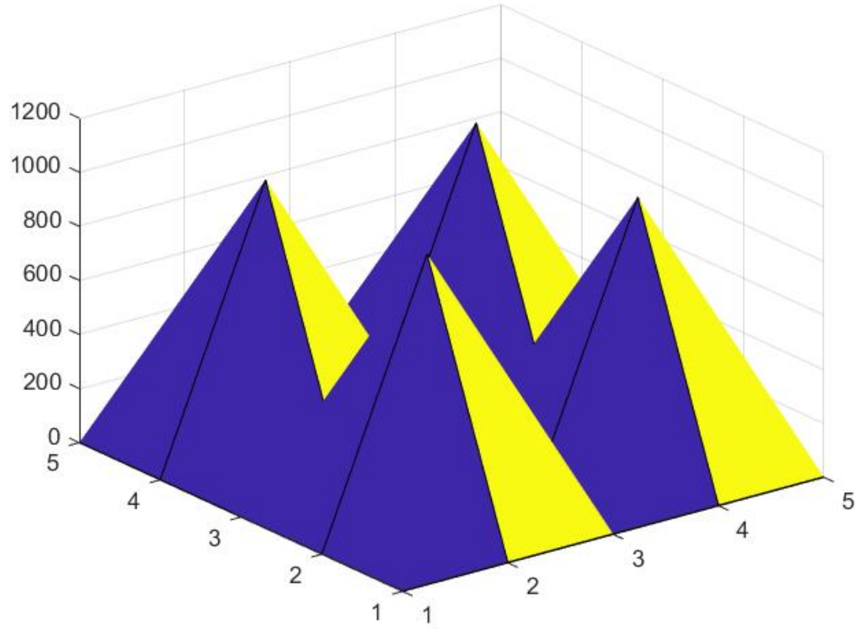


Figure 6.5: **Pressure distribution - Fastest algorithm - Static Image**

One of the way to use this algorithm safely is to keep an upper bound on the speed tailored to the user. But, this in turn affects the time-efficiency making it less suitable.

Accelerating algorithm

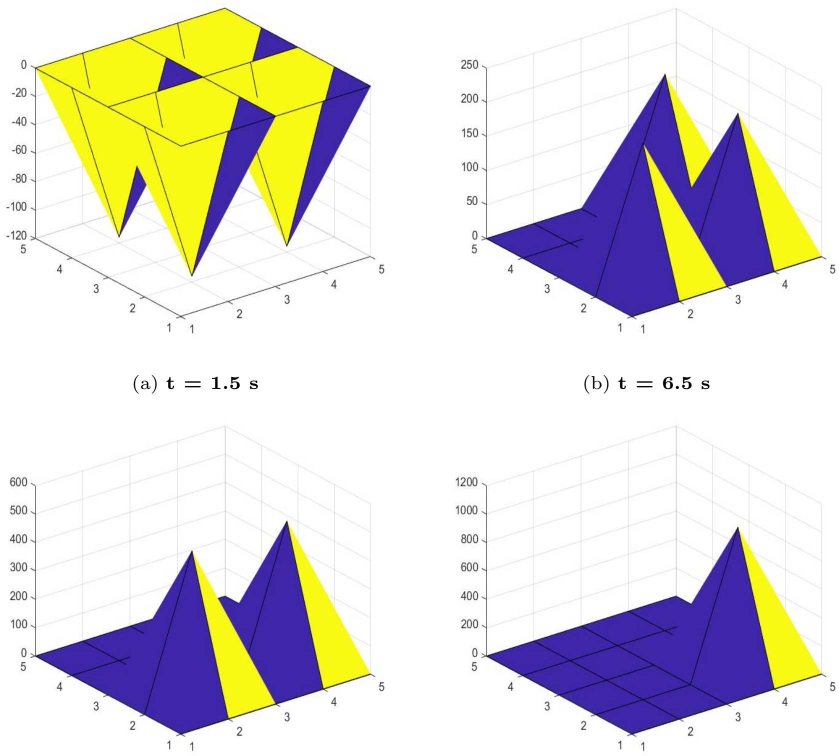
In section 5.2.2 we saw the working of the accelerating algorithm to create a static image 5.5. From figure 5.7, we can see that the total time has increased to 12 s, thus showing poor time efficiency.

Figure 6.6 shows the pressure distribution along time as the static image is created. The figure 6.6a shows negative pressure as the pressure exerted by hand is higher in this case. We can see the pressure increasing as the speed increases. In this case as well, as we can see from figure 6.6d, the pressure goes beyond 1000kPa which is much higher than data from results of the survey.

But, from the single pin survey feedback, the participants were comfortable when the pin starts slow and increases the speed. This is likely because anticipating the speed increase, the user may be lifting the hand, which makes the pressure ineffective. So, this can be a good algorithm for when the user is well informed. But still, the risks are high.

Decelerating Algorithm

Section 5.3.2 explained the working of the algorithm for the static image creation 5.5. Like the accelerating algorithm, this had a high refresh rate, which reduces



(a) **t = 1.5 s** (b) **t = 6.5 s**

(c) **t = 9.9 s** (d) **t = 12 s**

Figure 6.6: **Pressure distribution - Accelerating algorithm - Static Image**

the time efficiency.

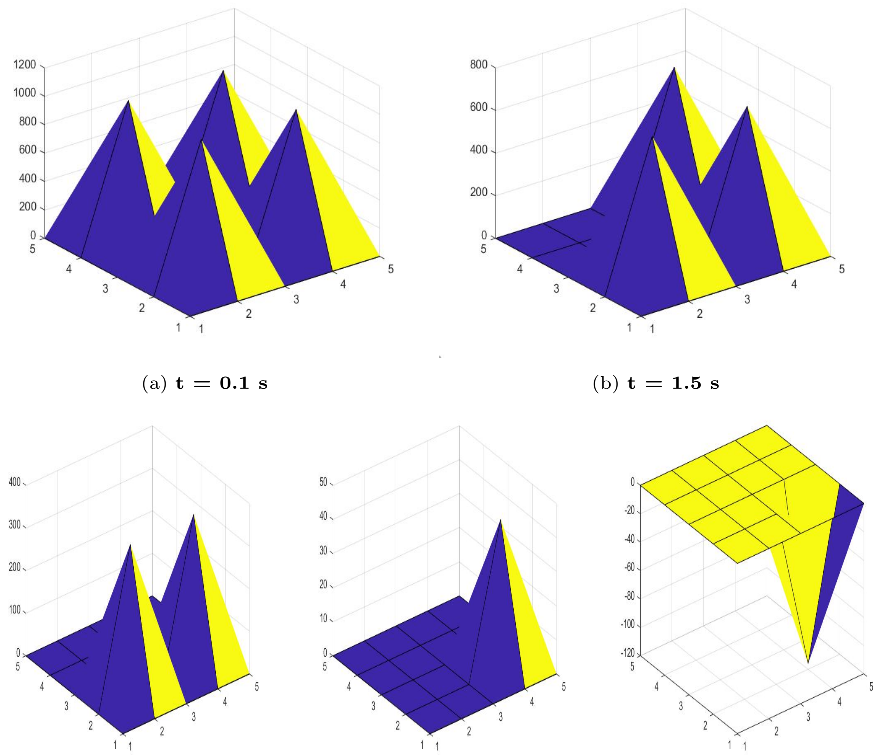
Figure 6.7 shows the pressure distribution at different times while creating the image. At time 0.1s, that is at the start itself, we have a very high pressure value which doesn't abide to the survey results. As time increase, this pressure can be seen to decrease. This is because of deceleration of the actuators. It reaches a negative value when the force exerted by hand is higher.

This is seen to be a really unsafe algorithm for static image creation due to the very high pressure at the start itself. So, this algorithm shouldn't be used.

Sea wave algorithm

Section 5.4.2 shows the static image creation example using the sea wave algorithm. This take 3 s to create the image which is much lesser compared to the accelerating and decelerating algorithms.

Figure 6.8 shows the pressure distribution when the actuators rise in a sea wave algorithm. This creates a uniform pressure of about 1192kPa. When comparing to the pressure results from survey in figure 6.4, the participants are able to tolerate more than 1000kPa of pressure when 4 actuators move. This means that, with the number of actuators being as high as 25, this should be



(a) **t = 0.1 s** (b) **t = 1.5 s**

(c) **t = 4.5 s** (d) **t = 9.1 s** (e) **t = 12 s**

Figure 6.7: **Pressure distribution - Decelerating algorithm - Static Image**

comfortable for all the users.

Though this algorithm performs worse than the fastest algorithm in terms of time, this algorithm is the best in safety and comfort. This algorithm can be good trade-off for the small increase in time.

But, another disadvantage of this algorithm is the high power consumption in operating all the motors to create any image. This may be drawback where applications aim for power-efficient displays.

Segmented sea wave algorithm

Section 5.5.2 outlines the working of the algorithm for creating the static image 5.5b. From figure 5.10, we can see that the time taken for this algorithm is same as the sea wave algorithm. This means time wise, it is as efficient as the sea wave algorithm.

Figure 6.9 shows the pressure distribution while creating the static image. Initially, its starts with a uniform pressure similar to the sea wave algorithm. This is because of the way the final image is chosen. Next, in figure 6.9b, we can see the pressure of about 1000kPa spread over 21 pins, and it reduces to 15 pin exerting pressure of about the same value and then only 9 pins exerting

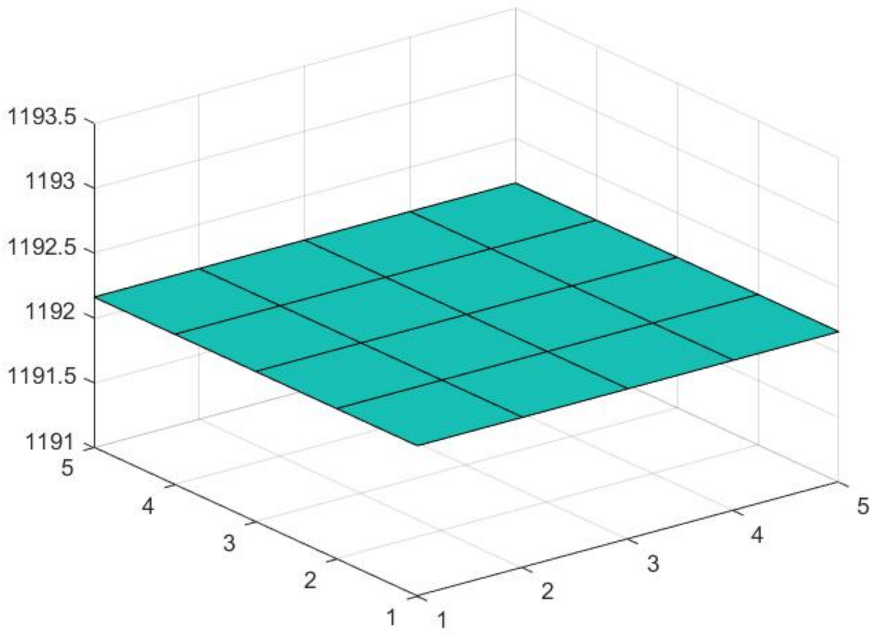


Figure 6.8: **Pressure distribution - Sea wave algorithm - Static Image**

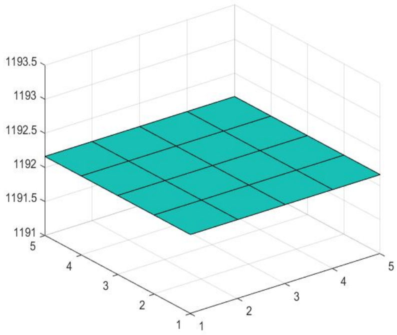
about 1000kPa pressure. Considering the results of survey where most users were comfortable with 1000kPa with 4 pins, it can be extended to say this will also be comfortable experience.

Comparing it to the sea wave algorithm, it can be seen that this is much more power efficient and is almost as comfortable. The small trade in comfort can be done for power-efficient applications.

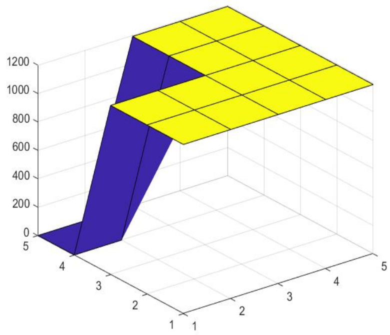
Speed vs area algorithm

Section 5.6.2 shows how the actuators work for creating the static image 5.11b. This is the most inefficient algorithm in terms of time efficiency. But this is application dependant. The time lag is mainly created by the single pin moving at a slow speed. For applications like showing the function of a sea wave or buildings which will have multiple pins travelling to create an image, this will have a better time efficiency.

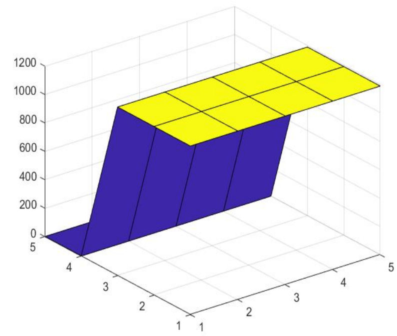
Figure 6.10 shows the pressure distribution for the example discussed earlier. We can see that in each case of 1 pin, 2 pins, 3 pins and 4 pins scenarios, the pressure values are within the limits that most participants from the survey were comfortable with. This is better in terms of safety and comfort when compared to the accelerating and decelerating algorithms.



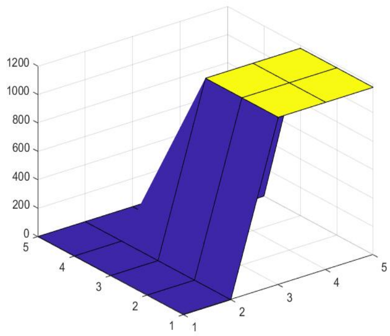
(a) **t = 0.1 s**



(b) **t = 0.5 s**



(c) **t = 0.9 s**



(d) **t = 1.2 s**

Figure 6.9: **Pressure distribution - Segmented sea wave algorithm - Static Image**

Summary

From the analysis of the static image application for the different algorithms, we can conclude these points:

1. The most comfortable and safe algorithm is the sea wave algorithm which also has a good time efficiency. But this is power inefficient.

2. The segmented sea wave algorithm provides almost the same level of comfort and time efficiency while providing a better power efficiency than the sea wave algorithm. This is the preferred algorithm for most static image applications. But this creates in-between images due to adjacent pins rising. This might be undesirable in some scenarios.

3. The speed vs area algorithm provides appropriate safety and comfort but has very low time efficiency.

4. The fastest, accelerating and decelerating algorithms are not advised since it can be less comfortable for the user.

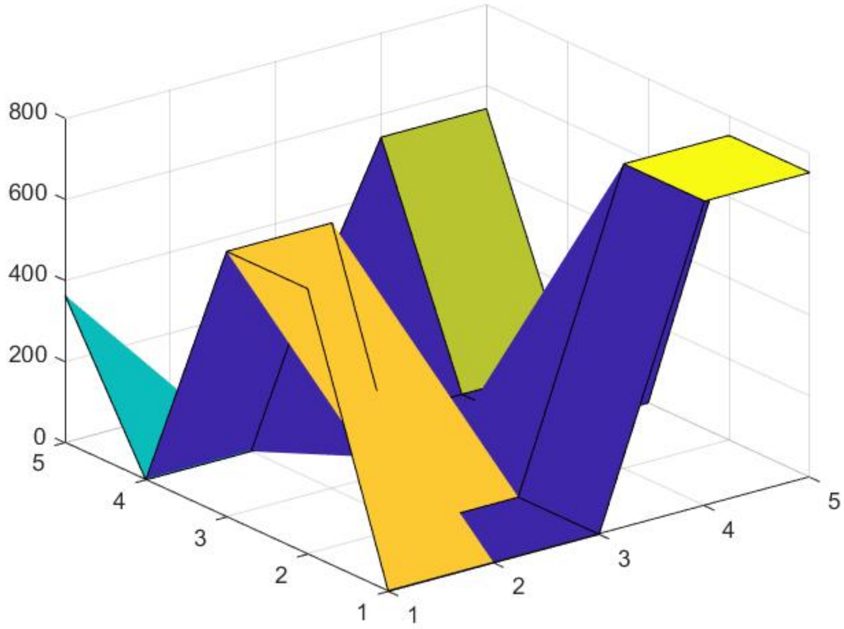


Figure 6.10: **Pressure distribution - Speed vs area algorithm - Static Image**

6.3.2 Dynamic image applications

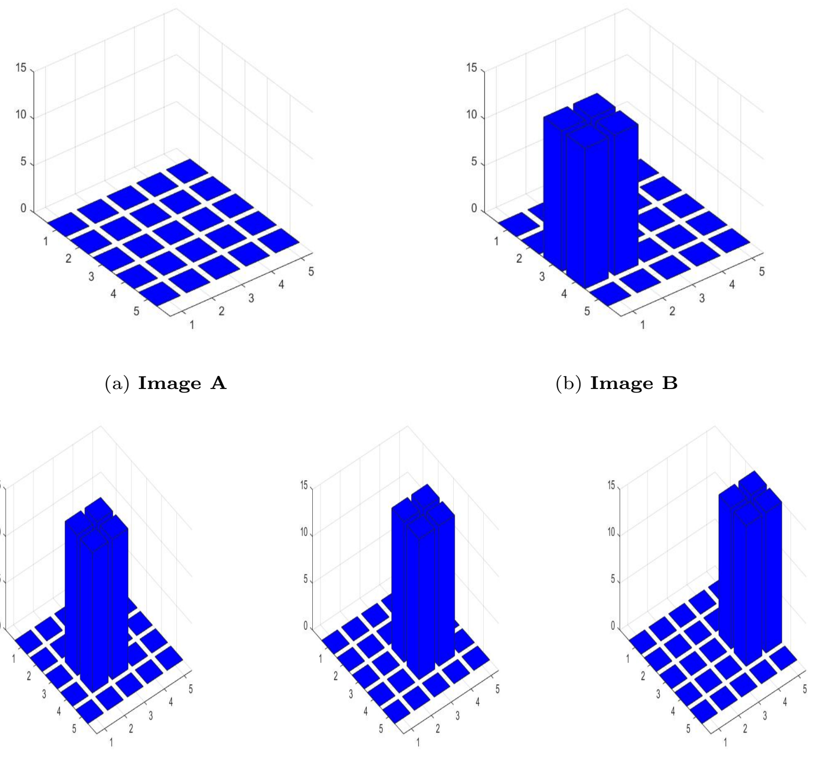
Dynamic image means that the image is changing constantly with time. Some of the applications for this are physical tele-presence, gaming interface and 2.5D camera like display to help guide the visually impaired.

In this section, we evaluate how each of these algorithms work for creating a dynamic image. Then we compare the pressure distribution chart for each with the survey results to understand which of the algorithms are safe and comfortable for the application. Based on these, we provide suggestions for choosing the best algorithm.

Dynamic image example

For evaluating the algorithms for efficiency, safety and comfort in dynamic image application, we look at the example for moving an object. In this example, we show how a 2×2 object will look to move from one end of the sub-module to another.

Figure 6.11 shows the example chosen with the images it will go through to reach the final image. The image 6.11a is the starting image and once we have the object in image 6.11b, we move it to image 6.11e through images 6.11c and 6.11d. Each sub-section will discuss how this example is implemented using different algorithms.



(a) **Image A** (b) **Image B**

(c) **Image C** (d) **Image D** (e) **Image E**

Figure 6.11: **Dynamic image example**

Fastest algorithm

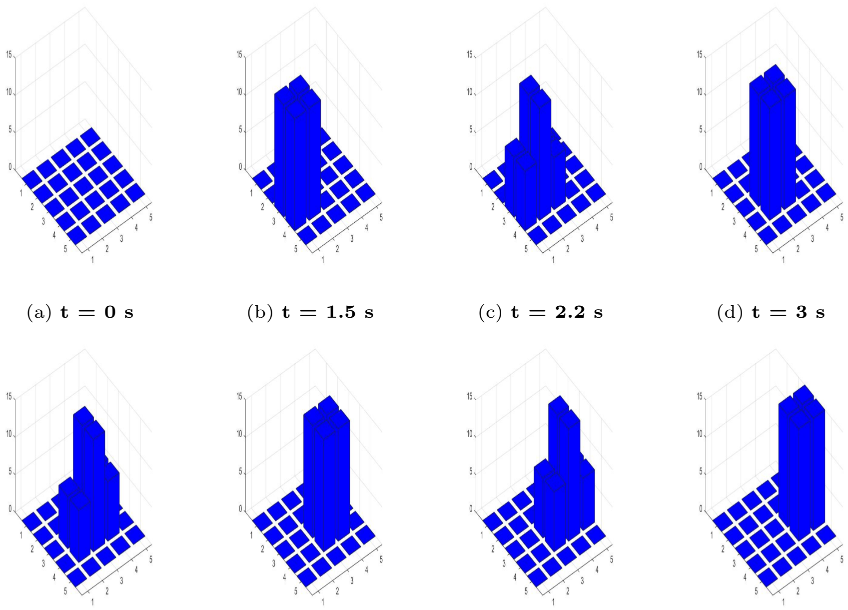
As discussed in chapter 5, fastest algorithm is the most basic algorithm with the least logic. So, to create a dynamic image, it moves from one image to another at the highest speed. Figure 6.12 shows the example 6.3.2 implemented using the fastest algorithm.

As can be seen from the figure, the total dynamic image takes 6 s to execute. There is a perfect transition from one image to another, thus showing this algorithm is, in general suitable for dynamic image creation. This is the fastest algorithm, but has its disadvantages when it comes to user safety and comfort. Figure 6.13 shows the pressure distribution curve when executing this algorithm.

As can be seen from the figure, with 4 pin movement, the pressure reaches more than 1000kPa and even with two pins, its close to 1000kPa. From the survey data, we can see that for the 2 pin set up, very few participants were comfortable with 1000kPa pressure. Thus, we can see that this is not a safe and comfortable algorithm for this application.

Since user dependent changes can be made to the algorithm, we can impose a maximum limit on the speed with this algorithm. This will affect the time efficiency, but will make it more comfortable for the user.

So, we can say that, with a limit on the maximum speed, this can be good



(a) **t = 0 s** (b) **t = 1.5 s** (c) **t = 2.2 s** (d) **t = 3 s**

(e) **t = 3.8 s** (f) **t = 4.5 s** (g) **t = 5.2 s** (h) **t = 6 s**

Figure 6.12: **Fastest algorithm - Dynamic image**

algorithm for dynamic applications.

Accelerating algorithm

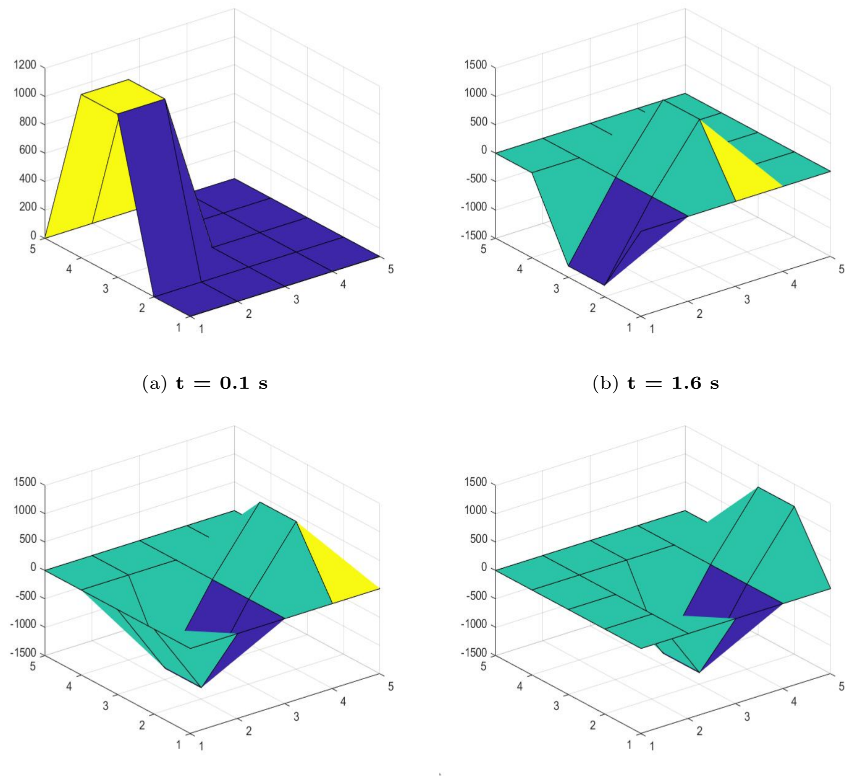
In chapter 5, we discussed how this algorithm works. Now we will see how it will perform for a dynamic application. The example in section 6.3.2 was implemented using the accelerating algorithm and its results are shown in figure 6.14.

From the image we can see that the dynamic movement is implemented without any hindrances in the image. This means that, in general, we can use this algorithm for dynamic image creation. It takes 42 s to create the entire dynamic image example. This is about 8 time worse performance compared to the fastest algorithm in terms of speed.

Figure 6.15 shows the pressure distribution when the algorithm is executed for the given example.

From the pressure distribution, we can see that in the beginning, the pressures are well within the limit of the survey result. But with time we can see that at 12 s, it exerts the maximum pressure of nearly 1200 kPa over 4 pins. From the survey data, we can see that for 4 pins, though most of the participants were comfortable with 1000 kPa, there were some participants who were only comfortable with 800 or 900 kPa. In later time as well, 2 pins exert about 1000 kPa, which majority of the participants were not comfortable with. Thus, this algorithm may not be ideal for the application scenario.

This can still be used with a cap on the maximum speed, but we have seen that the fastest algorithm can perform better with the maximum speed cap,



(a) **t = 0.1 s** (b) **t = 1.6 s**

(c) **t = 3.1 s** (d) **t = 4.6 s**

Figure 6.13: **Pressure distribution - Fastest algorithm - Dynamic Image**

thus rendering this algorithm unnecessary.

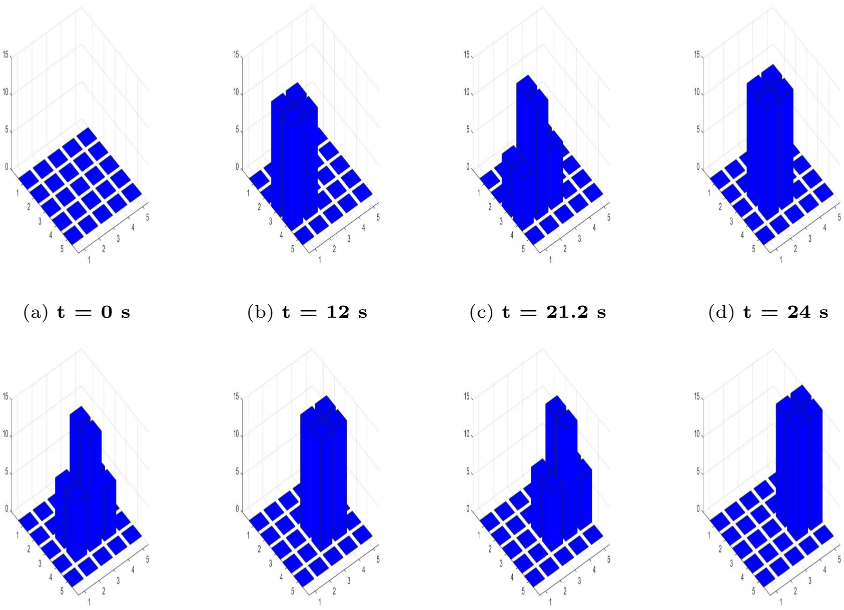
Decelerating algorithm

Chapter 5 also discuss the working of the decelerating algorithm. To understand its effect in dynamic applications, the example from 6.3.2 was implemented using this algorithm. Figure 6.16 show the working of this algorithm for the application.

From the figure, we can see that the time taken for the complete execution is 48 s, which is same as that of the accelerating algorithm. We can see that the clear dynamic image can be formed without the need of any in between images. This makes is suitable for dynamic applications.

The figure 6.17 shows the pressure distribution curve for the example. It looks very similiar to the accelerating algorithm with the pressure peaking at different instances, So, we can come to the same conclusion that this is not ideal algorithm for the application scenario.

And with a cap on the maximum speed, we can use this algorithm safely, but wouldn't be as beneficial as using the other algorithms.



(a) **t = 0 s** (b) **t = 12 s** (c) **t = 21.2 s** (d) **t = 24 s**

(e) **t = 32.4 s** (f) **t = 36 s** (g) **t = 44.4 s** (h) **t = 48 s**

Figure 6.14: **Accelerating algorithm - Dynamic image**

Sea wave algorithm

Chapter 5 discussed the working of the sea wave algorithm. This was seen as a very safe algorithm for the static image application. In this section, we discuss how it performs for dynamic application cases.

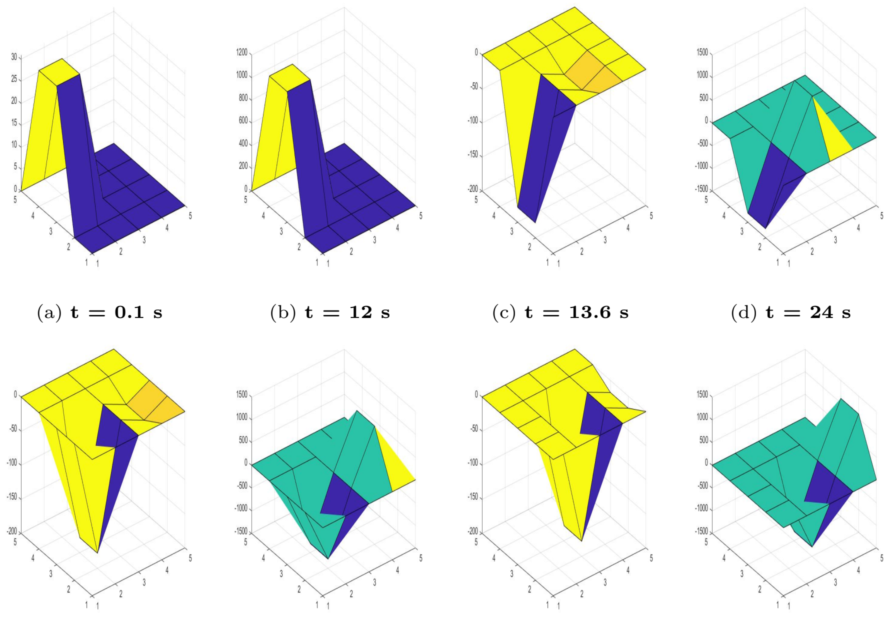
Figure 6.18 shows the working of this algorithm for the example mentioned in section 6.3.2.

The total time for execution of the complete image is 12 s, which is much better than the accelerating and decelerating algorithms. In this we can see that, the algorithm work by creating in between images as seen in figures 6.18b, 6.18e, 6.18h and 6.18k. This can disrupt the flow in which the dynamic image is visualized, as in this case, it is hard to understand it depicts moving of an object. So, this may not be an ideal algorithm for most of the dynamic applications. Also, as seen in the case of static image application, this is a high power consuming algorithm.

The figure 6.19 shows the pressure distribution chart for different timings during the execution of the algorithm.

From the pressure distribution chart in image 6.19a we can see a pressure of about 1200 kPa created across 25 pins. Though this value is high, the results of the multi-pin survey shows that most of the participants could tolerate this pressure when 4 pins are moving synchronously. This trend could be extrapolated to understand that this pressure will comfortable for most of the users.

Also, figures 6.19b, 6.19c, 6.19d and 6.19e shows about 1200 kPa pressure across 21 pins. The same extrapolation from the previous case applies here as well. So, in terms of safety and comfort, this algorithm performs best for the



(a) **t = 0.1 s** (b) **t = 12 s** (c) **t = 13.6 s** (d) **t = 24 s**

(e) **t = 25.6 s** (f) **t = 36 s** (g) **t = 37.6 s** (h) **t = 48 s**

Figure 6.15: **Pressure distribution - Accelerating algorithm - Dynamic Image**

application instance.

Though safety criterion is met, we can see that this is not an ideal algorithm for dynamic algorithm and thus not advisable for the application type.

Segmented sea wave algorithm

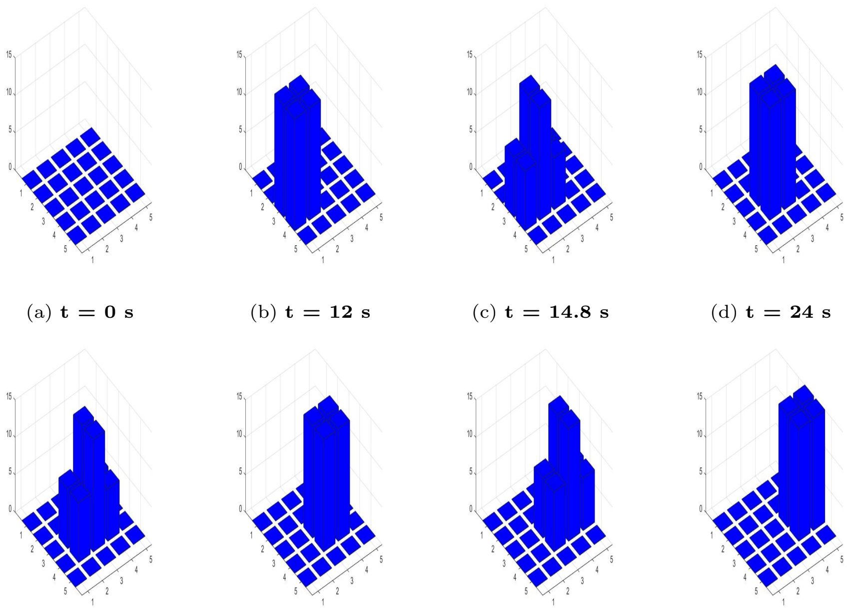
The segmented sea wave algorithm as discussed earlier is an extension of the sea wave algorithm to make it more power efficient. This section discusses its working and performance in dynamic application cases.

Figure 6.20 shows the segmented sea wave algorithm operation for the dynamic image creation example mentioned in section 6.3.2.

Time wise, the segmented sea wave algorithm performs similar to the sea wave algorithm for this application example. But, in terms of power, this algorithm does better than sea wave algorithm. This algorithm also creates in between images as seen in figures 6.20b, 6.20e, 6.20h and 6.20k. This can disrupt the flow of the dynamic image making the transitions unclear.

Figure 6.21 shows the pressure distribution at particular times in the previous example.

Figure 6.21a shows a pressure of 1200 kPa spread across 12 pins. This pressure was comfortable for most of the participants of the multi-pin survey with 4 pin setup, thus extending the understanding that it will be comfortable for 12 pins. Figures 6.21b and 6.21c shows pressures of about 1000 kPa across 10 pins. Same as the previous case, we can say that this will be comfortable for almost all the users. Figure 6.21d shows about 1000 kPa pressure across 6 pins. This, we



(a) **t = 0 s** (b) **t = 12 s** (c) **t = 14.8 s** (d) **t = 24 s**

(e) **t = 26.8 s** (f) **t = 36 s** (g) **t = 38 s** (h) **t = 48 s**

Figure 6.16: **Decelerating algorithm - Dynamic image**

can see is almost close to the survey result and few users might find this a bit uncomfortable.

Though a few pressure values were border line, this was still not as high as other algorithms, thus making it more safer. But, due to the intermittent images that will be created by this algorithm, this is not an ideal solution for dynamic applications.

Speed vs area algorithm

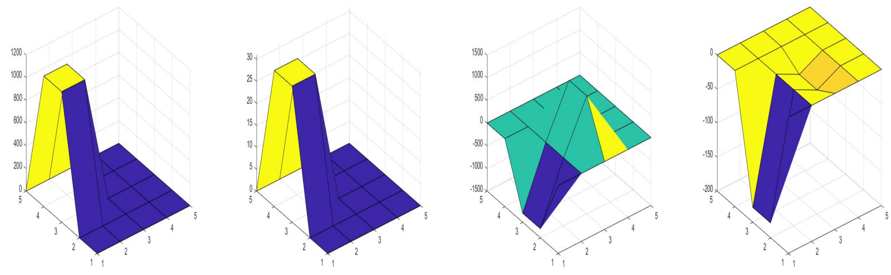
In this section, we see the dynamic image creation using the speed vs area algorithm and evaluate its performance. Figure 6.22 shows how the dynamic image from section 6.3.2 is created using this algorithm.

The total time taken to create this dynamic image can be seen as 25.8 s. This is much lesser than the accelerating and decelerating algorithm. The images do not show any disruption in the flow of creating a dynamic view. So, this is suitable for creating dynamic image applications.

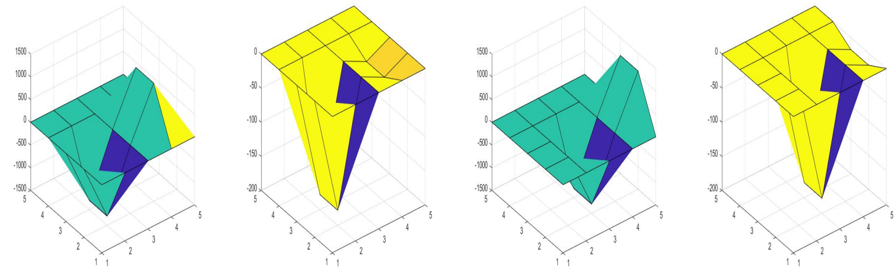
Figure 6.23 shows the pressure exerted chart when creating the dynamic image.

Figure 6.23a shows a pressure of about 600 kPa exerted over 4 pins. The multi-pin survey results show that all participants were very comfortable with this pressure for the 4-pin setup. Figures 6.23b, 6.23c and 6.23d shows about 500 kPa pressure created over 2 pins. From the results of 2-pin set up, we can see that all participants were comfortable with this pressure.

This algorithm has been seen to be very safe and comfortable for the application type. Though it performs worse than the fastest algorithm in terms of



(a) **t = 0.1 s** (b) **t = 12 s** (c) **t = 12.1 s** (d) **t = 24 s**



(e) **t = 24.1 s** (f) **t = 36 s** (g) **t = 36.1 s** (h) **t = 48 s**

Figure 6.17: **Pressure distribution - Decelerating algorithm - Dynamic Image**

time, that is a trade-off advised for a safe operation. Also, with the ability to create dynamic images, this is a highly recommended algorithm for the dynamic image application type.

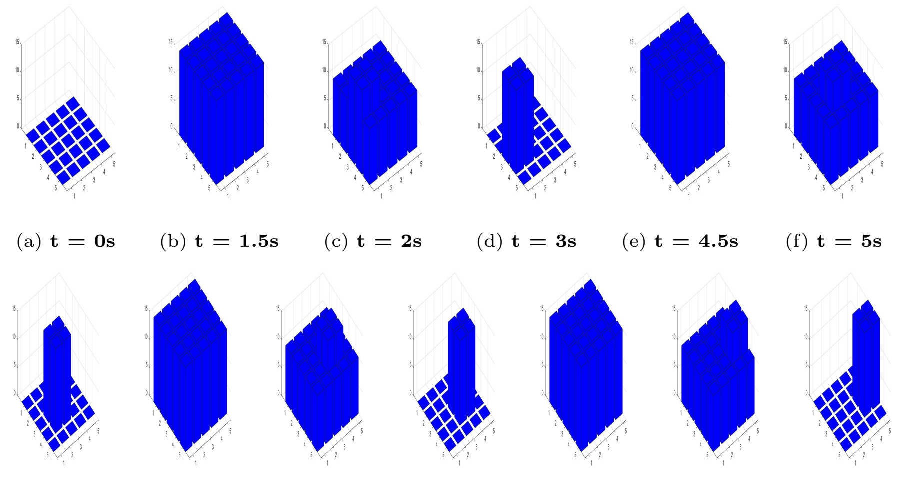
Summary

From the analysis of the dynamic image example, we can conclude the following points:

1. Speed vs area algorithm performs the best in terms of safety and comfort and is the advised algorithm for the application type.
2. The accelerating and decelerating algorithms though can be used for dynamic images, are not safe and comfortable.
3. The sea wave and segmented sea wave algorithms, though are seen to be safe and comfortable, cannot be used for dynamic applications due to the intermittent images they create, thus disrupting the flow.

6.4 Chapter Summary

In this chapter, we saw how the simulator was designed and its operation. We saw the method of evaluation used to evaluate the algorithms and we evaluated the algorithms for two different application types of creating static and dynamic

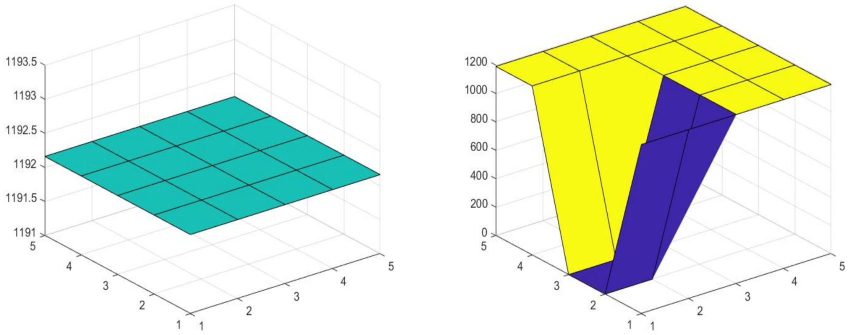


(a) **t = 0s** (b) **t = 1.5s** (c) **t = 2s** (d) **t = 3s** (e) **t = 4.5s** (f) **t = 5s**

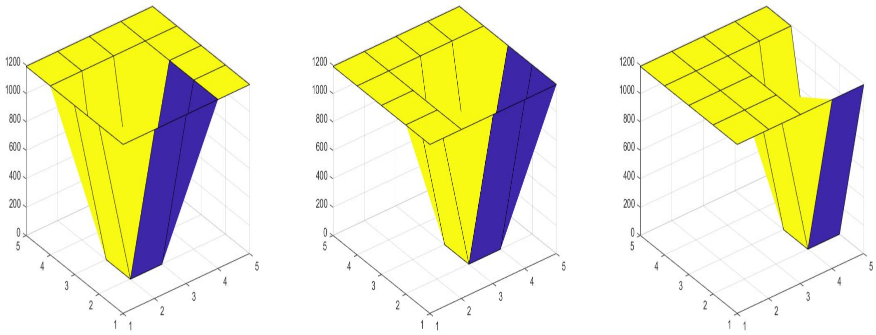
(g) **t = 6s** (h) **t = 7.5s** (i) **t = 8s** (j) **t = 9s** (k) **t = 10.5s** (l) **t = 11s** (m) **t = 12s**

Figure 6.18: **Sea wave algorithm - Dynamic image**

images. Based on this evaluation, the best algorithms for each application type were suggested.

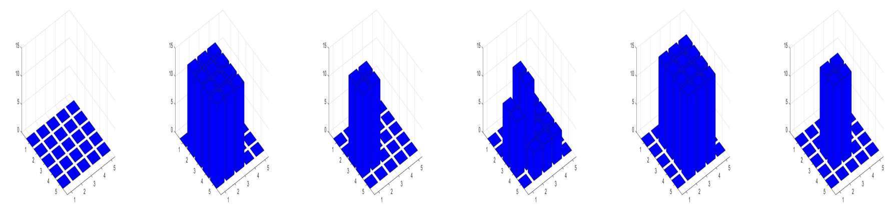


(a) **t = 0.1 s** (b) **t = 3.1 s**

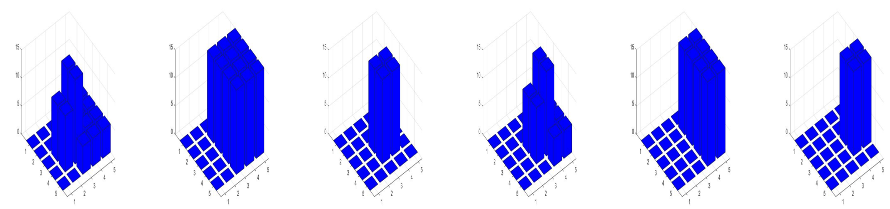


(c) **t = 6.1 s** (d) **t = 9.1 s** (e) **t = 12 s**

Figure 6.19: **Pressure distribution - Sea wave algorithm - Dynamic Image**

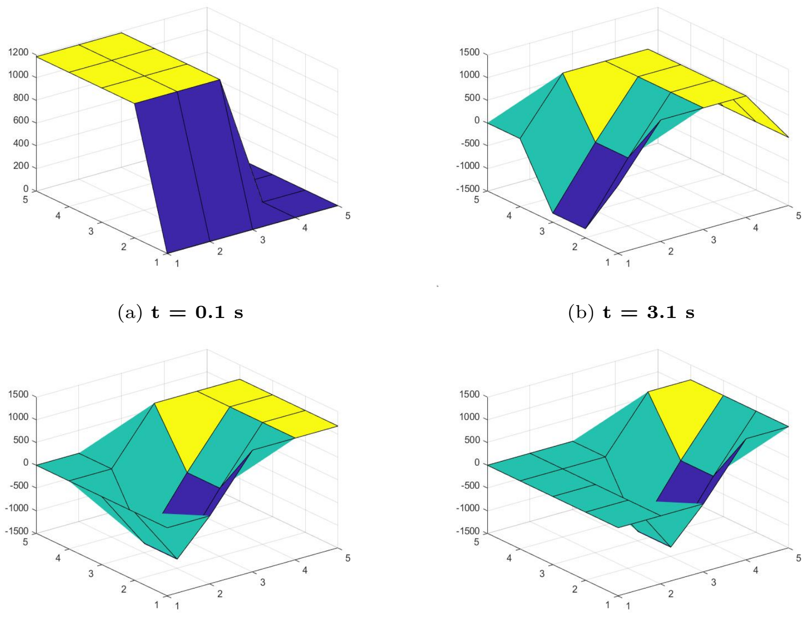


(a) **t = 0 s** (b) **t = 1.5 s** (c) **t = 3 s** (d) **t = 3.5 s** (e) **t = 4.5 s** (f) **t = 6 s**



(g) **t = 6.5 s** (h) **t = 7.5 s** (i) **t = 9 s** (j) **t = 9.5 s** (k) **t = 10.5 s** (l) **t = 12 s**

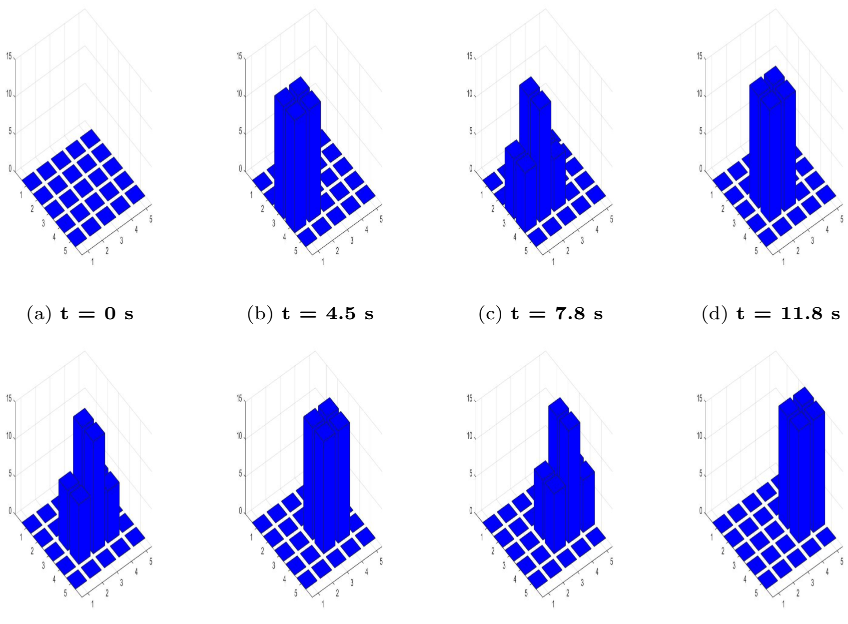
Figure 6.20: **Segmented Sea wave algorithm - Dynamic image**



(a) **t = 0.1 s** (b) **t = 3.1 s**

(c) **t = 6.1 s** (d) **t = 9.1 s**

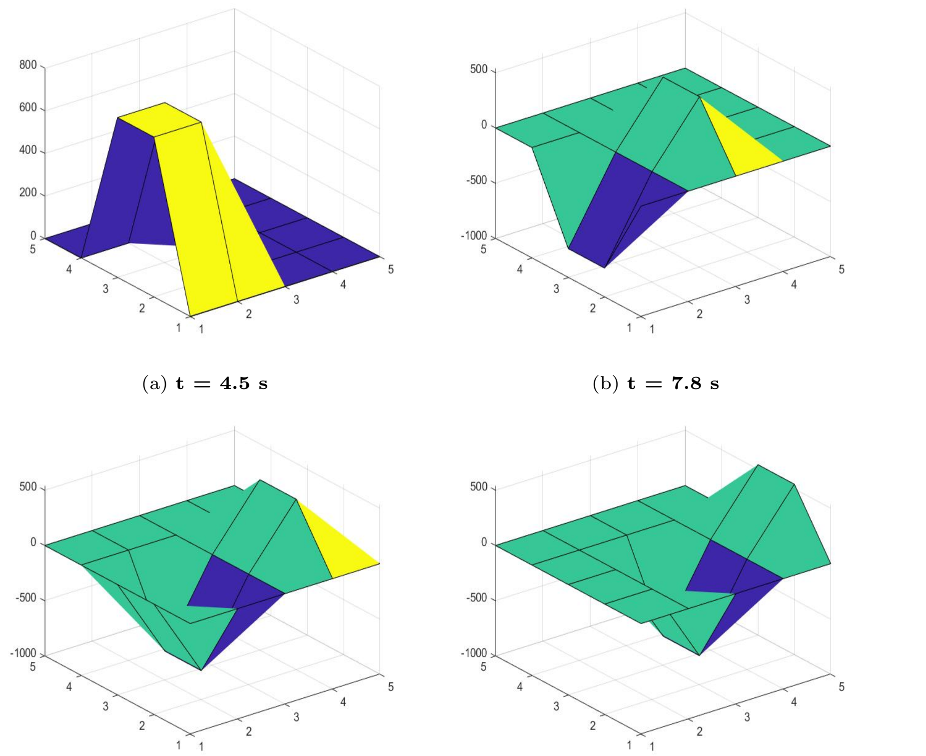
Figure 6.21: **Pressure distribution - Segmented sea wave algorithm - Dynamic Image**



(a) **t = 0 s** (b) **t = 4.5 s** (c) **t = 7.8 s** (d) **t = 11.8 s**

(e) **t = 15.3 s** (f) **t = 19.3 s** (g) **t = 22.3 s** (h) **t = 25.8 s**

Figure 6.22: **Speed vs area algorithm - Dynamic image**



(a) **t = 4.5 s**

(b) **t = 7.8 s**

(c) **t = 15.3 s**

(d) **t = 22.3 s**

Figure 6.23: **Pressure distribution - speed vs area algorithm - Dynamic Image**

Chapter 7

Conclusion and Future Work

In this chapter, we conclude the work done and present the major discussions and results of the thesis. Further, we propose some future works that can be done in this project.

7.1 Summary and Conclusion

In this thesis, we started with an idea for building a 2.5D display that can incorporate various applications. For this, we initially did a literature study to understand the existing state-of-the-art displays and understood the advantages and disadvantages of the existing designs. We understood that these existing models either focused on very high resolution or development of high end applications. There weren't any designs aimed at creating a safe and comfortable display at affordable price while providing high resolution. So, this was chosen as the area of focus.

Based on this, we decided to build our own display. We went on to do a feasibility study to learn about the hardware modules available in the market and its specifications. Using these, we saw how to build a scalable design and proposed two types of models for creating such a design. With the help of the DEMO team at Delft University of Technology, a cost estimate was made to build the proposed model.

We later saw that there wasn't any clear definition of what can be deemed as safe and comfortable to use a 2.5D display. So, we went on to understand and define safety and comfort through a series of experimental surveys. We created a prototype using 4 pins with which the experiments were conducted. Using the results of the survey, we defined a metric for safety and comfort for the designed model of the display.

In order to safely operate the displays, we needed a method of actuation which will be harmless to the user. So, we developed six different safe actuation algorithms. These algorithms varied in the way each actuator was moved to create an image and their speeds. But we needed a way to evaluate these algorithms to really understand if they are safe and which will be the best algorithm that can be used.

Initially, the plan was to build the hardware with funding from Eindhoven University of Technology. But, due to the covid-19 related actions, the labs had to be shut, creating a hindrance to this plan. So, instead we moved on to a simulator model which can mimic the design that we created. With the simulator that we built, we simulated different scenarios that will be specific to application instances and implemented them using the different actuation algorithms. From this, we created a pressure map for each time instance. This pressure map was compared with the results of the survey to understand how each algorithm performed. Best algorithms for each scenario was then proposed.

Combining the algorithms and the prototype design, we propose a completely safe and comfortable 2.5D shape display that the user can interact with. This was also seen to have comparable resolution to the state-of-the-art displays but at lower cost.

Some of the major conclusions of this project are as below:

1. For the design of the 2.5D shape display, a geared system consisting of a stepper motor and lead screw and nut is chosen as the actuator and a photoreflector is used as the sensor.

2. Of the designed safe actuation algorithms, the segmented sea wave algorithm and speed vs area algorithm are the advised options for static image creation based on the application instance.

3. For the dynamic image creation, the speed vs area algorithm is the recommended option.

7.1.1 Project shortcomings

Though the design showed good result, we are aware of some of the shortcomings of the project. These include:

1. Survey participant count - The more the number of people taking part in a survey, the better will be the result. But the situations pertaining to Corona did not allow us to expand the survey beyond the minimum requirement count.

2. Survey results - There are various situations that can affect how a person may feel comfortable to any object at a given point of time. These may be emotional state, surrounding temperature, or even the task the person was performing right before the interaction. Also, the way the person holds their hands and the area where the pins actually touches the hands, all these impact the comfort scale of the user. To further understand these, more specific surveys with higher participants will be required.

3. The results of the simulation follow ideal situations. In practice, there might be other various factors that will come into play such as how the pins will respond to hand pressure, any irregular sensor data, etc. These will require an actual prototype test to confirm the result.

4. The algorithms and the evaluation results are very specific to the hardware design we chose. This is not a universal solution, but a method of how this can be achieved.

7.2 Future work

The 2.5D shape displays are still an area in research. With evolving hardware and production methodologies, these displays can also be improved for performance, cost and applications. Some of the aspects that can be improved in the future for this specific project are as suggested:

1. The actual production of the hardware. This will let us to perform the simulator based tests on the hardware to fine-tune the results and get results much closer to the needs of the user.

2. One of the advancing field for the 2.5D display is the building of applications aimed for the visually impaired. They are more sensitive to the environment meaning their comfort levels will be different compared to others. Performing the survey with visually impaired participants can help in creating hardware that can be used for such applications.

3. Building various applications using the chosen algorithms to understand how it performs in real life scenarios.

Bibliography

[1] 3d shape display · bruno yos. [online] Available at: `hhttp://bdehoyos.me/projects/display`, last accessed Sept. 2020.

[2] Bmx-biometal, 2020. [online] Available at: `https://www.biometal.biz/bmx`, last accessed Sept. 2020.

[3] Cjp single acting pin cylinder, 2020. [online] Available at: `https://www.smc.eu/uk-ua/products/cjp-pin-cylinder-single-acting-spring-return~39713~cfg`, last accessed Sept. 2020.

[4] Comfort - definition, 2020. [online] Available at: `https://dictionary.cambridge.org/dictionary/english/comfort`, last accessed Sept. 2020.

[5] Ghost display surfaces, 2020. [online] Available at: `http://www.ghost-fet.com/home/`, last accessed Sept. 2020.

[6] Micro stepper motor, 2020. [online] Available at: `https://nl.aliexpress.com/item/32970448136.html?spm=2114.search0302.3.108.139379b9WNDI0f&ws_ab_test=searchweb0_0,searchweb201602_0,searchweb201603_0,ppcSwitch_0&algo_pvid=0f4156f9-3de8-46bc-9bab-f86ab4403929&algo_expid=0f4156f9-3de8-46bc-9bab-f86ab4403929-15`, last accessed Sept. 2020.

[7] Safety - definition, 2020. [online] Available at: `https://dictionary.cambridge.org/dictionary/english/safety`, last accessed Sept. 2020.

[8] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. Grand challenges in shape-changing interface research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–14, New York, NY, USA, 2018. Association for Computing Machinery.

[9] Sharma Anirudh and Motiyar Rahul. Zixel : A 2.5d graphical tactile display system. In *Proceedings of the 2011 ACM Siggraph conference*, New York, NY, USA, 2011. Association for Computing Machinery.

[10] Lennart Blanken. *Learning and Repetitive Control for Complex Systems With Application to Large-Format Printers.* PhD thesis, Eindhoven University of Technology, 2020.

[11] Louis Cohen, Lawrence Manion, and Keith Morrison. *Research methods in education*. Routledge, 6 edition, 2009.

[12] Meredith Gall, Walter Borg, and Joyce Gall. Educational research: An introduction. *British Journal of Educational Studies*, 32, 2003.

[13] Ewa Garniec. cubeme, 2020. [online] Available at: `https://garniec.wordpress.com/2008/07/06/cubeme/`, last accessed Sept. 2020.

[14] Hiroo Iwata, Hiroaki Yano, Fumitaka Nakaizumi, and Ryo Kawamura. Project feelex: Adding haptic surface to graphics. In *Proceeding of SIGGRAPH2001*, pages 469–476, 2001.

[15] D. Leithinger, S. Follmer, A. Olwal, and H. Ishii. Shape displays: Spatial interaction with dynamic physical form. *IEEE Computer Graphics and Applications*, 35(5):5–11, 2015.

[16] Daniel Leithinger, Sean Follmer, Alex Olwal, and Hiroshi Ishii. Shape displays: Spatial interaction with dynamic physical form. *IEEE Computer Graphics and Applications*, 35:5–11, 2015.

[17] C. Loconsole, D. Leonardis, M. Gabardi, and A. Frisoli. Braillecursor: an innovative refreshable braille display based on a single sliding actuator and simple passive pins. In *2019 IEEE World Haptics Conference (WHC)*, pages 139–144, 2019.

[18] TT Motors. 6mm coreless planetary gear micro dc motor - buy planetary gear micro motor,coreless motor,6mm dc motor product on alibaba.com, 2020. [online] Available at: `https://www.alibaba.com/product-detail/6mm-coreless-planetary-gear-micro-dc_60827650395.html?spm=a2700.7724838.2017115.155.6fbf52ebte9ONZ`, last accessed Sept. 2020.

[19] Masashi Nakatani, Hiroyuki Kajimoto, Dairoku Sekiguchi, Naoki Kawakami, and Susumu Tachi. 3d form display with shape memory alloy. In *ICAT*, 2003.

[20] Dan Overholt. The matrix: A new musical instrument for interactive performance. *ICMC*, 2001.

[21] Vera ZasÃPÃ Ariza and Mauricio SantÃs Chaves. Interfaces hãpticas: sistemas cinestãsicos vs. sistemas tãctiles. *Revista EIA*, pages 13 – 29, 2016.

[22] Ivan Poupyrev, Tatsushi Nashida, Shigeaki Maruyama, Jun Rekimoto, and Yasufumi Yamaji. Lumen: Interactive visual and shape display for calm computing. In *ACM SIGGRAPH 2004 Emerging Technologies*, SIGGRAPH '04, page 17, New York, NY, USA, 2004. Association for Computing Machinery.

[23] Raf Ramakers, Johannes Schöning, and Kris Luyten. Paddle: Highly deformable mobile devices with physical controls. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, page 191–192, New York, NY, USA, 2014. Association for Computing Machinery.

[24] U. Sekaran and Roger Bougie. Research methods for business: A skill building approach (5th edition). *International Journal of Information Technology and Management - IJITM*, 2009.

[25] Alexa F. Siu, Eric J. Gonzalez, Shenli Yuan, Jason B. Ginsberg, and Sean Follmer. Shapeshift: 2d spatial manipulation and self-actuation of tabletop shape displays for tangible and haptic interaction. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery.

[26] Luke Vink, Viirj Kan, Ken Nakagaki, Daniel Leithinger, Sean Follmer, Philipp Schoessler, Amit Zoran, and Hiroshi Ishii. Transform as adaptive and dynamic furniture. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, page 183, New York, NY, USA, 2015. Association for Computing Machinery.

[27] K. Zhang and S. Follmer. Electrostatic adhesive brakes for high spatial resolution refreshable 2.5d tactile shape displays. In *2018 IEEE Haptics Symposium (HAPTICS)*, pages 319–326, 2018.

[28] David Černý. David cerny - uplne oficialni stranka, 2020. [online] Available at: `http://www.davidcerny.cz/start.html`, last accessed Sept. 2020.