FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Exploring Pseudo-Haptics for object compliance in VR

Carlos Lousada

Mestrado em Engenharia Informática e Computação

Supervisor: Prof. Daniel Mendes

Second Supervisor: Prof. Rui Rodrigues

July 19, 2023

Exploring Pseudo-Haptics for object compliance in VR

Carlos Lousada

Mestrado em Engenharia Informática e Computação

July 19, 2023

Abstract

Virtual Reality (VR) and the development of virtual environments have allowed users to experience and access virtual 3D worlds and simulate reality. VR applications in health, education, and entertainment areas require immersive and interactive systems. To achieve that, haptic feedback is one of the key elements, but most devices that provide it are difficult to access and costly. For this reason, exploring alternatives is increasingly important. In that context, the concept of Pseudo-Haptic feedback - using visual and auditory stimuli to create an illusion or modify the perceived haptic feedback - emerges as both an addition to (passive) haptic devices and as a complete alternative.

In this work, we started by exploring Haptic concepts and particularly Pseudo-haptic feedback techniques while focusing on ones that function as an alternative to haptic devices. During that research, we discovered a couple of techniques that showed promising results but had not been explored nor adapted in detail to a Virtual Reality environment for the perception of compliance (the opposite of stiffness).

We proposed to combine and adapt the visual elements of those techniques. We designed a solution that was based mainly on visual feedback created by the physical deformation of objects, adapted from the Mass-Spring-Damper model. We included the utilization of hand-tracking software as the interface for the user to interact with the deformable, joined with an algorithm of inverse kinematics that triggers upon contact, to increase the level of realism and enhance the perception of compliance.

The results were very positive, as most of the participants in the research confirmed that they were able to easily identify different levels of compliance with high confidence while considering that they were inserted in a realistic and immersive environment provided by the approach proposed. We also identified an influence of the object's scale on the perception of this property in a couple of situations, as the participants affirmed that they tend to identify smaller objects as more compliant.

Keywords: Haptic Feedback, Pseudo-Haptic Techniques, Virtual Reality, Virtual Object Properties, Compliance, Stiffness, Deformation, Mass-Spring-Damper model, Inverse Kinematics, Hand-Tracking

ACM Classification:

Human-centered computing → Human-computer interaction (HCI) → Interaction Paradigms → Virtual Reality

Human-centered computing → Human-computer interaction (HCI) → Interaction Devices → Haptic Devices

Resumo

A Realidade Virtual (VR) e o desenvolvimento de ambientes virtuais permitiram aos utilizadores experienciar e aceder a mundos virtuais 3D e simular a realidade. As aplicações de VR nas áreas da saúde, educação e entretenimento exigem sistemas imersivos e interactivos. Para o conseguir, o feedback háptico é um dos elementos-chave, mas a maioria dos dispositivos que o fornecem são de difícil acesso e dispendiosos. Por esta razão, a exploração de alternativas é cada vez mais importante. Neste contexto, o conceito de feedback pseudo-háptico - utilizando estímulos visuais e auditivos para criar uma ilusão ou modificar a perceção do feedback háptico - surge como um complemento aos dispositivos hápticos (passivos) mas também como uma alternativa completa.

Neste trabalho, começámos por explorar conceitos hápticos e, em particular, técnicas de feedback pseudo-háptico, concentrando-nos naquelas que funcionam como uma alternativa aos dispositivos hápticos. Durante essa investigação descobrimos um par de técnicas com resultados promissores, mas que não tinham sido exploradas nem adaptadas em pormenor para um ambiente em Realidade Virtual de forma a atingir perceção de conformidade (o oposto de rigidez).

Propusemos combinar e adaptar os elementos visuais dessas técnicas. Concebemos uma solução que se baseava principalmente no feedback visual criado pela deformação física de objectos, adaptado do modelo Mass-Spring-Damper. Incluímos a utilização de software de rastreio da mão como interface para o utilizador interagir com o deformável, juntamente com um algoritmo de inverse kinematics que é acionado após o contacto, para aumentar o nível de realismo e melhorar a perceção de conformidade.

Os resultados foram bastante positivos, uma vez que a maioria dos participantes na investigação confirmou que era facilmente capaz de identificar diferentes níveis de conformidade com elevada confiança, considerando também que estavam inseridos num ambiente realista e imersivo proporcionado pela abordagem proposta. Também identificámos uma influência da escala do objeto na perceção desta propriedade num par de situações, uma vez que os participantes afirmaram que tendem a identificar objectos mais pequenos como mais conformes.

Keywords: Feedback Háptico, Técnicas Pseudo-Hápticas, Realidade Virtual, Propriedades de um objeto virtual, Conformidade, Rigidez, Deformação, Modelo massa-mola-amortecedor, Cinemática inversa, Rastreamento de mão

ACM Classification:

Computação centrada no ser humano → Interação homem-computador (IHC) → Paradigmas de interação → Realidade virtual

Computação centrada no ser humano → Interação homem-computador (IHC) → Dispositivos de interação → Dispositivos Hápticos

Acknowledgements

The writing of this dissertation and this entire academic journey has been a complete roller coaster. It has marked my life profoundly, and the completion of this chapter would not have been possible without all the support and love from various people who were a massive part of my success.

First and foremost, I would like to express my deepest gratitude to dissertation supervisors, Professors Daniel Mendes and Rui Rodrigues, who not only provided critical guidance and shared their exceptional expertise during this work but also showed tremendous availability and enjoyment in working with me. They were both essential to give continuous feedback, advice, and incentive to move forward during the entire process of this study.

To all of my colleagues, I would like to thank you for being a part of my whole academic life, and it is my pleasure to have shared all of the moments of this chapter with all of you.

I am also incredibly grateful to my friends, especially my close university friend group Auntdulce, for all the help, company, and camaraderie you showed during the difficult and unprecedented times we had to live through during the pandemic. Thank you for all the experiences and memories we built together, all of the late-night conversations and games we played, and, of course, congratulations to each and every one of you for the academic success we have achieved together.

I am particularly thankful to three of my friends who were absolutely crucial in different chapters of my academic journey. To my friend Filipe Recharte, who was always there for me, no matter if it was to discuss a university topic or simply to discuss and listen to my issues. To my friend David Rocha, who accompanied me on the incredible experience of the Erasmus+ program, where we shared great moments and learned about independent life together. And finally, to my friend Diogo Almeida, who was crucial during the development of this research, as we shared workspace and ideas, discussed several topics, arranged our experiments together, and helped each other whenever needed.

To all of my remaining friends, who have allowed me to embark on incredible experiences with them and share deep conversations, parties, and other remarkable memories that will forever stay with me.

Last but not least, I am deeply touched by the support of my family, particularly my parents and older brother. You have been supporting me my entire life and encouraging me to achieve all of my life goals ever since I was a child. I am forever in debt to you for all of your constant love, help, and overall presence in my life. Thank you for always being there for me. I love you deeply.

And to any other people who have been a valuable part of my life that I did not mention above in detail, thank you for shaping my life and allowing me to become the person that I am today.

Carlos Lousada

“The best way to predict the future is to invent it”

Alan Kay

Contents

List of Figures

List of Tables

Abbreviations

2D	2 Dimensions / 2 Dimensional
3D	3 Dimensions / 3 Dimensional
HMD	Head Mounted Display
VR	Computer-Aided Design
XR	Cross Reality / Extended Reality
HCI	Human-Computer Interaction
C/D	Control/Display
IK	Inverse Kinematics
FEM	Finite Element Method
PBD	Position-Based Dynamics
CCD	Cycle Coordinate Descent
FABRIK	Forward and Backward Reaching Inverse Kinematics
DLS	Damped Least Squares

Chapter 1

Introduction

The concept of Virtual Reality (VR) has been looked into for a long time. Currently, VR is defined as a simulated (virtual) experience that uses hardware such as 3D Head-mounted displays (HMD) and pose-tracking devices to offer the user higher levels of immersion and interaction than the normalized on-screen virtual experiences.

Virtual Reality today allows its users to be immersed in a detailed and realistic environment by enabling the simulation of physical touch, interaction with virtual entities, and movement. This was achieved through investments from several companies (such as HTC, Meta, Sony, and Valve), which have allowed this area to develop considerably regarding both software and hardware. Some fundamental changes included merging the main functionalities required from a display and head-tracking devices into a single device as well as the addition of VR controllers to the kit.

The capabilities of such an industry have been targets of increasing attention for its potential and applicability in several different areas. Studies have shown that Virtual Reality can bring advantages to professional work and personal life improvements. These benefits mainly apply to professional areas that require interaction with 3D content, such as architecture, content creation, and health, as well as leisure activities, such as gaming and video-content consumption.

1.1 Motivation

Haptic feedback is an essential factor when it comes to achieving the mentioned levels of immersion and interaction. Haptic feedback refers to the use of touch to communicate with the user. Examples of this range from a simple vibration on a mobile device to highly complex and detailed force applied on a haptic suit. When it comes to VR, this concept is responsible for increasing immersion during the interaction with virtual entities and movement by simulating touch.

The area of Haptic Feedback has been under investigation for a long time, especially the way it could affect interactions in technological devices. It has recently been increasingly explored for Virtual Reality applications to match the uprising of their capabilities and popularity.

1.2 Problem

When it comes to tracking physical touch and providing haptic feedback, the common devices are either simple controllers or excessively costly tracking and force-feedback devices.

For this reason, the perception of haptic feedback and consequent immersion and realism for an average user - a user that owns only the basic VR equipment (an HMD and the corresponding trackers and controllers) and does not have the possibility or desire to acquire extra hardware - still has much room to improve and is worth exploring.

Additionally, as far as we know, the proposed alternatives to provide realistic haptic feedback via haptic devices are still very costly, tough to build, and, for that reason, very hard to access as an average user. Besides, most of these devices are developed with a particular problem in mind, reducing their applicability for a wider variety of applications.

With this in mind, a new concept emerged: Pseudo-Haptics. To the best of our knowledge, this concept was first formally used in 2000 [31], where it was defined as a combination of a passive force feedback device with visual feedback to simulate realistic haptic cues. Using techniques that do not entirely rely on haptic devices to provide haptic feedback means that their average cost is lower and that their implementation is simpler and broader, thus increasing the availability for an average user. For this reason, further exploration of techniques based on Pseudo-Haptics becomes more appealing.

In the last couple of decades, studies on pseudo-haptic techniques have been reasonably positive and have been explored as ways of solving particular issues. This means that each study usually focuses on the perception of one property of a virtual object (for example, the weight of an object [41]) or of a specific environment (for example, underwater drag forces [25]). However, many of these techniques have not been updated to be used in 3D rather than in 2D or still make use of mouse cursors, which is not as viable in VR applications.

1.3 Proposed Work

The way that pseudo-haptic techniques have been explored in recent years seems to ignore one of the main advantages of their usage: to solve a wider variety of issues in more straightforward and cheaper ways and thus be incorporated into a broader range of applications.

After looking deeply into the pseudo-haptic techniques that have been explored for virtual environments and virtual reality specifically, we found there to be a lack of exploration for the perception of compliance. Most techniques that explore this property are used in 2D applications and/or with mouse cursors. This concept and the existing explorations will be further explained in the next chapter.

For this reason, this study will focus on investigating the possibility of adapting the current techniques to measure compliance. Additionally, we will also explore if we can find a technique that can be applied to a virtual environment and positively influence immersion, realism, and level of interaction, while using mainly visual stimuli, thus increasing availability.

1.4 Document Structure

In the Related Work chapter, we will review the current state of the art in haptic feedback and, more specifically, pseudo-haptic techniques for VR. We will begin by discussing essential haptic concepts and disambiguating them, using relevant references. We will then examine the challenges and limitations of these techniques and the directions for the research we will pursue. Subsequently, in Chapter 3, we will introduce the proposed approach and explore in further detail the proposed technique and the implementation details. It will also discuss the development and architecture used in the prototype. Following that, in Chapter 4, we will discuss the evaluation of our technique, describe the tests performed and analyze the results, discuss them, and draw conclusions. Finally, in the Conclusion chapter, we will outline the overall work scope, point out the main conclusions from the study, and describe possible future work.

Chapter 2

Related Work

Haptic feedback, also known as haptics, refers to the use of the sense of touch to communicate with the user to provide information regarding something in a virtual or physical environment. In virtual reality (VR), haptic feedback can be simulated through haptic devices such as controllers that provide force feedback, touch-sensitive surfaces, or gloves. By giving users the impression that they are interacting with virtual objects and environments in the same way they would in the real world, haptic feedback can improve the realism and immersion of VR experiences.

However, haptic devices can be costly and challenging to develop, which has limited their widespread adoption in VR. As a result, researchers have explored alternative techniques for simulating haptic feedback in VR, known as pseudo-haptic techniques. Pseudo-haptic methods rely on visual or auditory feedback, or a combination of both, to create the illusion of touch in a VR environment. These techniques have the potential to be more available and cost-effective than traditional haptic devices, making them a promising area of research for VR applications.

2.1 Key Haptic Concepts

As the area of Haptic feedback kept evolving and being studied, several concepts related to it soon emerged, sometimes making the objective distinction between them less clear. The idea of Pseudo-Haptics itself was also explored in two main different ways: the starting purpose - the combination of passive-haptic devices with visual feedback to create an illusion that better represents natural environment properties; and the usage of mainly or exclusively visual techniques to provide the user with haptic cues that allow them to perceive environment properties. For this reason, we will begin by discussing essential haptic concepts and disambiguating them, using relevant references.

2.1.1 Visual Haptics

Visual haptics, also known as visuo-haptics, in most articles (as used by Sandor et al. [42]), is defined as a concept that englobes all systems that use haptic devices and visual stimulation to allow the users to perceive haptic properties.

However, in a few articles (as used by Nomoto et al. [38]), it is defined as a system that aims to simulate haptic cues based on visual stimuli and illusions (also referred to as Pseudo-haptics 2.1.4). In this work, we will not consider the latter and instead use the expression "visual-based pseudo-haptics" or synonyms, for that matter.

2.1.2 Active Haptics

Active haptic interfaces are computer-controlled physical interfaces that provide objective feedback to the user by changing the device's properties. These devices are typically very costly to design and produce, expensive to acquire as a user (with the simplest ones costing over 200$ and the complex ones over 5000$), and have very specific applications. Active haptic interfaces can simulate the force required to manipulate virtual objects or provide haptic feedback when interacting with virtual environments [35, 9].

Examples of active haptic devices include force feedback controllers, haptic gloves, and touch-sensitive surfaces. These devices can provide a wide range of haptic sensations, including kinesthetic, tactile, and thermal feedback: Kinesthetic haptic feedback refers to the body's sense of movement or force. It can be simulated through techniques such as force feedback, which applies force to the user's body or limbs. Examples of kinesthetic haptic feedback include the sensation of weight or resistance when lifting or manipulating virtual objects. Tactile, haptic feedback refers to the sense of touch or texture on the skin. It can be simulated through techniques such as vibrotactile feedback, which uses vibration to stimulate the skin, or touch-sensitive surfaces that provide tactile feedback when touched. Examples of tactile, haptic feedback include the sensation of roughness or smoothness when touching virtual objects. Finally, thermal haptic feedback refers to the sense of temperature on the skin. It can be simulated through temperature-sensitive surfaces or devices that apply heat or cold to the skin. Examples of thermal haptic feedback include the sensation of warmth or coldness when interacting with virtual objects or environments.

However, as mentioned previously, active haptic devices can be challenging to design and costly to produce, making their widespread adoption limited. In addition, these devices may be limited in their applicability to specific tasks or environments.

The most standard examples of active haptics are force feedback controllers such as the Oculus Touch controllers or the HTC Vive controllers, which provide simple vibro-tactile feedback when interacting with virtual objects and environments in a VR environment.

A more complex example is haptic gloves, such as the CyberGrasp (Figure 2.1), HaptX, and MANUS Haptic VR gloves. These provide in-depth and detailed kinesthetic and tactile haptic feedback depending on the interaction with virtual objects and environments in VR.

The most advanced active haptic system is haptic feedback suits, such as the Teslasuit and the bHaptics TactSuit, which, similarly to the gloves, also provide in-depth and detailed kinesthetic and tactile haptic feedback but over the entire user's body.

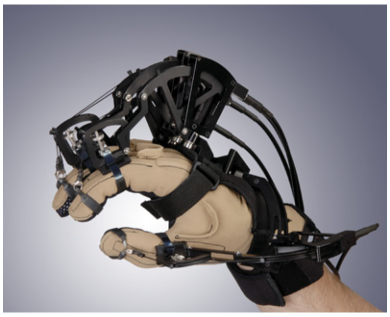


Figure 2.1: The CyberGrasp haptic gloves from CyberGlove Systems.

2.1.3 Passive Haptics

Passive haptic interfaces have some similarities with active haptic devices, with the main difference being that they are not computer-controlled. This means that the interface provides feedback to the user but does not depend on the application or system it uses. Passive haptic interfaces can simulate the feel of different materials or surfaces, allowing users to touch and manipulate virtual objects while perceiving sensation. Passive haptic interfaces may be less expensive and more widely applicable than active haptic devices. Examples of passive haptics are haptic devices and surfaces that use arrays of passively activated tactile actuators or piezoelectric sensors and mechanical or electrical mechanisms to provide haptic feedback to the user's skin. An exciting example of passive haptics is the multi-finger device called FlexiFingers explored by Achibet et al. [1], which can be visualized in Fig. 2.2. This device constraints each finger individually and produces elastic force feedback to simulate different stiffness levels when interacting with virtual objects. Another example is the Pit room explored by the University of North Carolina [24].

2.1.4 Pseudo Haptics

Pseudo-haptics techniques have been explored in several ways, the main one being the combination of passive haptic devices with visual feedback to create an illusion that better represents natural environment properties and the use of mainly or solely visual techniques to provide haptic cues that allow users to perceive environment properties. According to Lécuyer et al. [31], pseudo-haptics can be defined as the combination of a passive force feedback device with visual feedback to simulate realistic haptic cues. Following this definition, the exploration of the FlexiFingers mentioned above would also fit into this category.

Figure 2.2: Different iterations of the FlexiFingers device. [1]

On the other hand, pseudo-haptics techniques are also explored as techniques that aim to provide haptic feedback or the illusion of touch in a virtual reality (VR) environment without using traditional haptic devices such as force feedback devices. In this case, pseudo-haptics techniques rely on visual feedback, auditory feedback, or a combination of both to create the illusion of touch. These techniques have the potential to be further available and cost-effective than traditional haptic devices, making them a promising area of research for VR applications. In this work, we will mainly explore the second type of pseudo-haptic techniques, but without disregarding the possibility of using combination techniques. The following section will find more detailed state-of-the-art for Pseudo-Haptic techniques.

2.2 State-of-the-art in Pseudo-Haptic techniques

As mentioned previously, one of the first formal definitions of pseudo-haptics was provided by Lécuyer et al. [31], who defined it as a combination of a passive force feedback device with visual feedback to simulate realistic haptic cues. Since then, a wide range of pseudo-haptics techniques has been developed and explored for different types of environments (2D, 3D, Virtual Reality, and Augmented Reality), including methods that do not require a combination with haptic devices.

That being said, for the sake of this work, we will focus on the possibility of adapting promising technologies and algorithms that do not require haptic devices to a Virtual Reality 3D environment.

In order to do that, we need to explore what are the different types of Pseudo-Haptic techniques, what environments they were developed and tested for, and what properties they give feedback about. After conducting this study, we will focus on one specific property and formulate a possible solution that matches our interests and goals. During our research, we divided the techniques into different categories that will be explained below: perspective distortion techniques, texture remapping and deformation techniques, sound localization and binaural rendering techniques, and touch redirection and object remapping techniques.

2.2.1 Perspective distortion

Perspective distortion techniques use changes in the perspective of the virtual environment or objects that dissociate from what's happening in the physical world to give the user the illusion of haptic feedback. This is the most explored and suitable type of technology since it can be implemented in several ways to give the user the illusion of different properties. It is deeply associated with the concept of Control/Display ratio ("The ratio between the amplitude of movements of the user's real hand and the amplitude of movements of the virtual cursor is called the Control/Display ratio.", Dominjon et al. [16]). Still, its definition has varied according to the application that it has been adapted for, particularly in the way that it is calculated, as well as what it's applied on (in a VR application, it could be the user's hand while in a 2D application, it would be the cursor). With this being said, and following the division based on target haptic property in Ujitoko

Figure 2.3: An illustration of the Control/Display Ratio manipulation utilized in the experiment. The user is shown lifting an optically tracked cube with the corresponding virtual representation, the movements of which are a fraction of the user's movements. (Taken from [41])

et al.'s [45] survey on Pseudo-Haptics, the related work examples for this type of technique will also be analyzed separately.

Starting with weight, Samad et al. [41] explored the possibility of manipulating the virtual hand's position when compared to the physical hand in order to achieve the illusion of perception of weight. Besides, to the best of our knowledge, it was the first to provide a quantification of the range that can be used in the Control/Display ratio function without disrupting the sense of presence. Moreover, this study could prove promising as the experiments were conducted using objects that have an approximate weight of a widespread VR controller, thus allowing the perception of haptic properties in VR with minimal extra devices (Fig. 2.3). Other works on Control/Display for weight perception include Dominjon et al.'s study [16], which was one of the first to test this technique in a Virtual environment and concluded that the utilization of this technique was able to modify the perceived mass of the objects strongly. Based on this study, Palmerius et al. (2014) [39] evaluated the usage of merely visual feedback compared to a mix of both visual and haptic and purely haptic, which concluded that both the mix and simply visual had positive results in manipulating the sense of weight.

The following property described in the survey is compliance. Compliance is a broad concept, as it denotes measuring a material's ability to deform or bend under an applied load (in other words, it is the opposite of stiffness). This is one of the main properties explored by the techniques of texture remapping and deformation. One of the first examples of exploring distortion techniques was in a study also recognized as one of the earlier definitions of pseudo-haptics mentioned previously [31]. In this article, two different experiments were conducted using a passive spring and a piston, in which the users were asked to push their thumb against the piston. A virtual version of the setup was displayed on the screen, and it was dynamically animated in order to mimic reality. After that, a distortion of the user's thumb would be computed based on the applied force, allowing different stiffness values to be perceived. As this exploration proved successful in evaluating the visual dominance of haptic perception, further studies on this technique surged. An example of that is Tatezono et al.'s study [43], which adopted the same approach explained previously but using the PHANToM haptic interface. A diagram that shows the displacement mapping in that work can be seen in Figure 2.4.

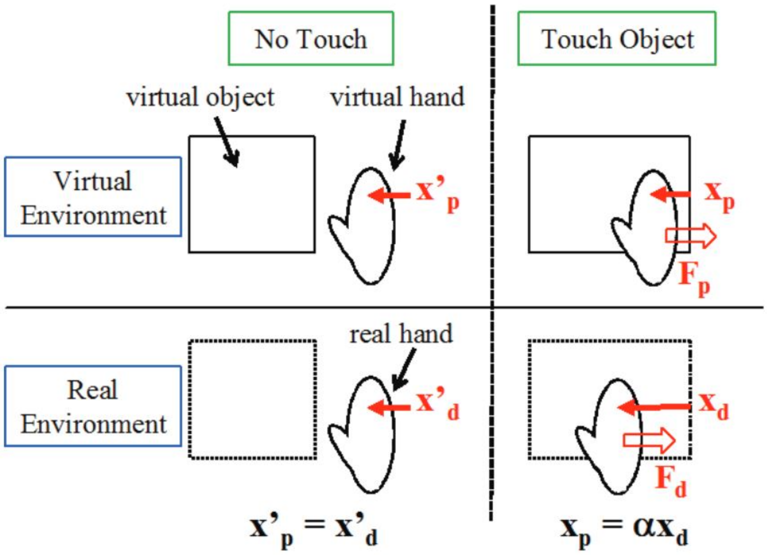


Figure 2.4: The application of a Control/Display ratio formula to the user's hand for the perception of compliance. (Taken from [43])

Weiss et al. [49] also proposed using Control/Display ratio to manipulate the way a user perceives stiffness based on their virtual hands. These studies, among others, use, however, extra haptic devices that would not be fitting for the goals proposed to explore in this dissertation. As for studies of this type of technique that do not use extra haptic devices, they instead explore the usage of a mouse by Kumar et al. [30] or a touch surface by Ridzuan et al. [40] and in non-VR environments, which makes them less relevant as well.

Moving on to another property, the usage of this type of technique for friction was also explored. Again, in the previously mentioned study by Lécuyer et al. [31], the property of friction was also explored by analyzing the user's response when asked to maintain the velocity of a cube. The adaptation in the feedback force coming from the device (due to the increase of the user's applied pressure on the ball), combined with the visual feedback on the velocity (the decrease in the speed of the object on screen), allowed users to perceive differences in friction. Most other explorations of kinetic friction include the usage of touch surfaces (such as the study by Narumi et al. [37] and one by Ujitoko et al. [48]), which makes them less relevant for this dissertation. The same applies to studies of static friction, which focused on the stick-slip phenomenon using touch surfaces (such as a work by Costes et al.[13], and one by Ujitoko et al. [47]).

Lastly, this type of technique also explored the property of roughness. The main application of the studies conducted for this property was to perceive bumps and holes. That started with Lécuyer et al.'s [33] exploration, in which the mouse's position would be distorted and adjusted according to the slope of the texture based on its height map. Another study by Lécuyer et al. [34] also explored the possibility of changing the cursor's speed and size based on similar properties. Other articles then followed in exploring similar situations (such as one by Argelaguet et al. [3], and one by Ujitoko et al. [46]), but most of them used 2D environments and/or made use of a cursor. The exceptions are Hannig et al.'s [22] studies, which used vibro-tactile feedback - thus reducing its relevance for this dissertation, as well as another work by Hannig et al. [21] that explored the use of Control/Display ratio to test the boundaries of its usage for this property.

2.2.2 Texture remapping and deformation

Texture remapping and deformation involve rendering visual patterns or textures onto virtual objects to give the user the illusion of touch. This technique can simulate the surface of a wide range of materials. Previous explorations of this type of technique were initially conducted by Argelaguet et al. [4] applied to simple 2D textures, and later by Ban et al. [7] for a 3D environment.

In the first one, 2D images representing an object of a particular material can be deformed with the click of the cursor. In the prototype presented, the rendering of the texture on the material will be recalculated and deformed according to the coefficient of elasticity, the points of contact, and the force (associated with the number and duration of clicks) (Fig. 2.5).

As for the second one, a similar effect was evaluated, but on a 3D object rather than a 2D image. Additionally, both of these works investigate the possibility of modifying the interaction interface. While in Argelaguet et al.'s [4] work, the alteration of the cursor itself does not prove relevant to this dissertation, the 3D exploration by Ban et al. [7] also investigates the possibility of deforming the virtual hand. The image processing process can also be visualized in the figure in that article (Fig. 2.6).

Another example of a texture remapping technique was measured by Ho et al. [23] by applying it to the perception of temperature. It used the mapping of the basic colors of red and blue in both a virtual object and the user's virtual hands in order to perceive slight temperature variations.

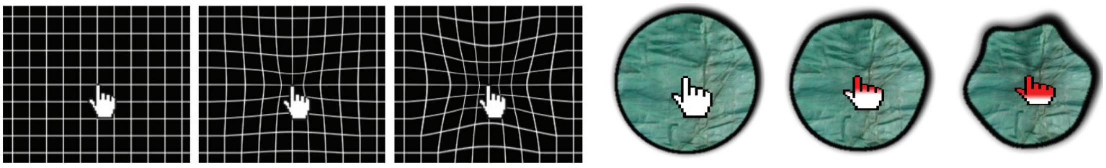


Figure 2.5: Elastic Image simulation. Left, animation steps for the proposed image-based deformation as the virtual pressure exerted by the user increases. Right, animation steps of the simulation of an Elastic Image with additional visual feedback.(Taken from [4])

2.2.3 Sound localization & binaural rendering

There is a scarcity of articles that explore sound-based techniques in the area of pseudo-haptic feedback since they do not allow the perception of any haptic property on their own.

Sound localization and binaural rendering techniques use changes in volume, frequency, timing, and location or direction of sound in order to provide the user with the illusion of haptic cues. For example, using sound cues to simulate movement or the presence of an object or an environmental property can give the user information about the environment he's inserted in and the objects around him. Further exploitation of these changes and techniques that involve the left and right ears can be used, for example, in a crash simulation.

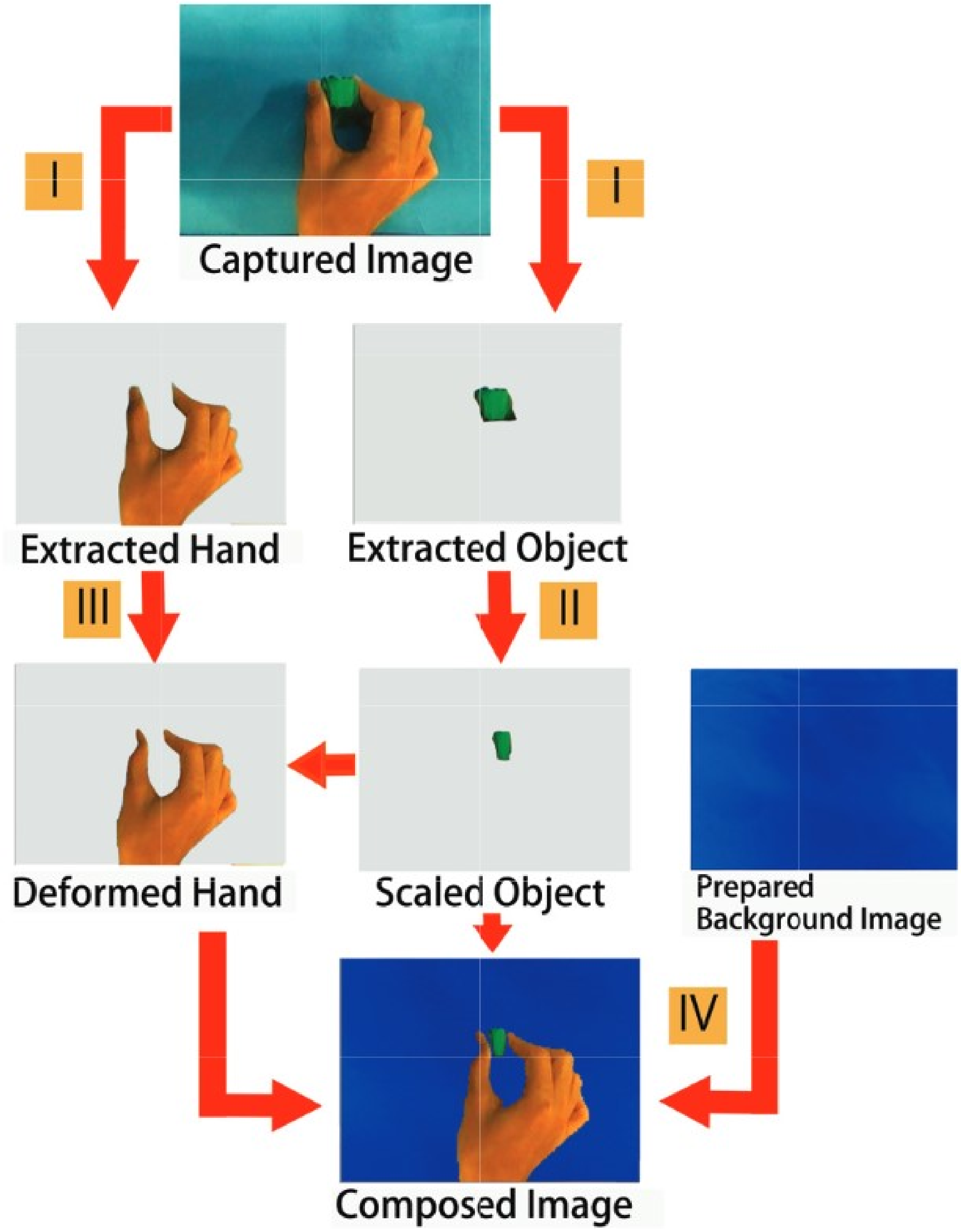


Figure 2.6: Image processing method for modifying the stiffness. (Adapted from [7])

As mentioned previously, these techniques are scarcely explored in the area of pseudo-haptics, but they have been studied in the area of multimodal feedback. The exploration of frequency and different sounds in order to aid in the perception of properties such as roughness, stiffness, and overall environmental immersion are included in studies like [2, 19, 27, 11].

2.2.4 Touch redirection & Object remapping

Even though the primary goal of this dissertation is to use the smallest amount of extra haptic devices, other techniques worth exploring include ones that use simple objects or widespread cheap hardware. Among these, the ones that better poked our interest are touch redirection and object remapping explorations.

Touch redirection is a technique that uses visual and/or auditory cues to guide the user's hand to a virtual location where they can perceive haptic feedback rather than providing haptic feedback directly to the user's hand. This technique can create the illusion of touch or contact with virtual objects or surfaces in a VR environment by leading the user to touch a device that provides a physical force or vibration to the user's hand. This technique is sometimes connected to the usage of Control/Display ratio (perspective distortion).

Object remapping is a technique that visually maps an object with fixed physical properties differently, altering the user's perception of the corresponding virtual object's shape, size, or other physical properties. This can be done through various means, such as changing the visual appearance of the virtual object and/or using sound to indicate the object's physical properties. It allows the user to perceive different virtual objects using the same physical thing.

Both of these techniques are deeply connected, and their previous explorations are also related. One of the most complex studies on these techniques was performed by Han et al. [20] as several experiments were conducted and different redirection and remapping formulas were evaluated. Figure 2.7 contains a simple diagram that shows how the hands and objects are processed. Other simpler studies include Kohli et al.'s [28] and Kohli et al.'s [29], which explore the remapping and redirection of passive haptic boards to perform simple clicking tasks, and a study by Azmandian et al. [6], which uses a single physical cube to allow the user to build virtual structures by stacking multiple virtual cubes.

2.3 Discussion

According to the information in the previous subsection, we defined a structure to classify the techniques in the mentioned types. We used the classification utilized in Ujitoko et al.'s [45] survey for the target haptic material properties.

Using this research, we can evaluate the main strengths and weaknesses of each of the mentioned technique types, as well as distinguish which haptic properties can be perceived according to the previous work provided.

Starting with the type of texture remapping and deformation, it is clear that it is not suitable to perceive a wide variety of different properties. From the work we found in that area, we realized

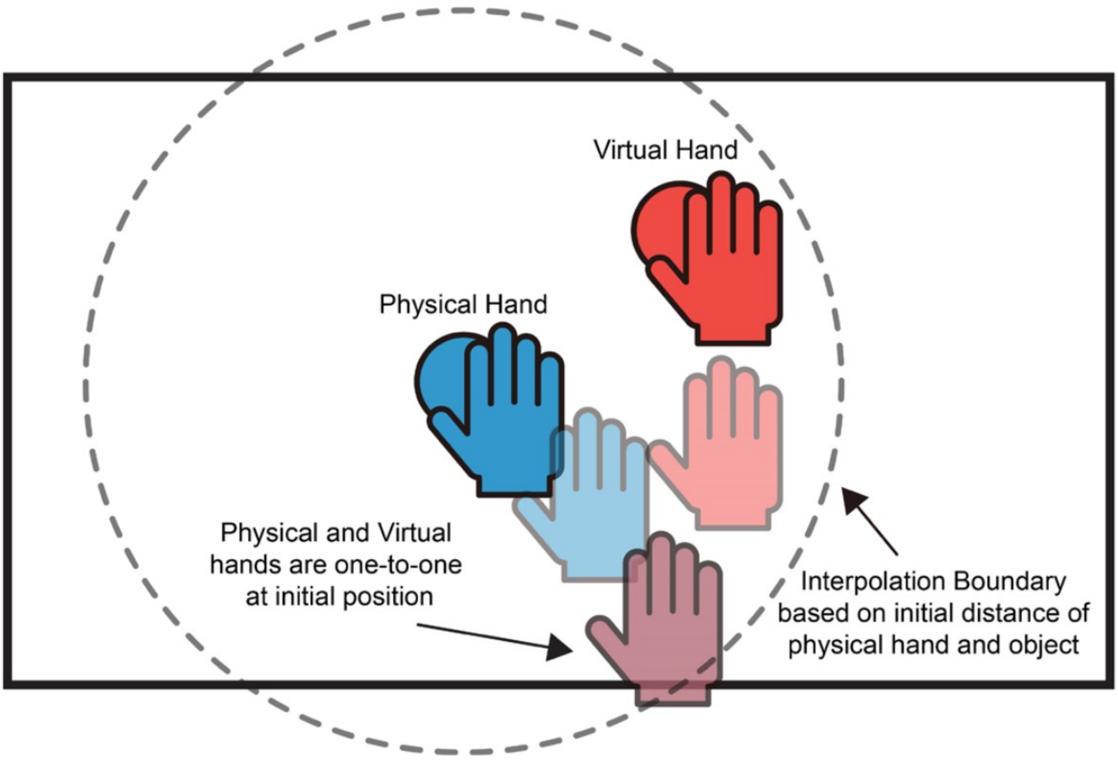


Figure 2.7: A top-down diagram demonstrating interpolated reach shows the physical hand and object in blue and the virtual hand and object in red. The dotted circle has a radius equal to the initial distance between the physical hand and the object, and it denotes the region where interpolation is in effect. Outside this boundary, a one-to-one mapping is used. (Taken from [20])

that it focused on exploring the properties of compliance and temperature, but there is a lack of research applied to VR applications. The high computational load and complexity of real-time remapping and deformation algorithms for 3D environments could be the main reasons for this scarcity. However, the research on this type of technique has achieved positive results employing a primary focus on visual elements rather than direct haptic feedback.

As for perspective distortion techniques, they have been deeply explored in a large number of studies in the area of pseudo-haptics. It is clear that they can be employed as a way to enhance the perception of several material properties, such as weight, compliance, friction, and roughness. However, most approaches of this type have utilized additional haptic devices (active, passive, or both) or input devices that are not as fit for VR environments (such as a mouse or a touchscreen).

When it comes to sound-based techniques, we can determine that they can only work as an addition to other techniques. This means that they are not able to induce the perception of a haptic property on their own. For that reason, the exploration of these techniques, to the best of our knowledge, has not been included directly in the area of pseudo-haptics but rather in the area of multimodal feedback. They do, however, have the capacity to enhance the performance of other techniques in the perception of a wide variety of properties as well as in providing environmental information.

Finally, regarding touch redirection and object remapping techniques, they always require the usage of an extra object or device to function. Even though they are able to provide information regarding most material properties, their accuracy is highly dependent on the quality and capabilities of the device it is combined with.

Based on this information, which is compiled in Table 2.1, we can see that to achieve the goals of this dissertation, the texture remapping and deformation techniques look promising, as they achieved satisfying results using relying mainly on visual feedback. Besides, there is a lack of exploration of their potential for VR environments. Additionally, perspective distortion techniques can also be of interest, as they have proven themselves in the enhancement of the perception of several properties. However, these techniques make use of either haptic devices or input systems that are not suitable for VR.

With this in mind, we got particularly intrigued by the possibility of exploring the perception of compliance in VR using a combination of both of these techniques. We focused specifically on adapting and improving the ideas proposed by Argelaguet et al. [4] and Ban et al. [7], with the primary goal of providing a similar effect to the deformation presented by the first work, while also using the idea of modifying the interaction interface (the user's hands) as proposed in the second work, using mainly visual feedback.

Table 2.1: Pseudo-Haptic techniques, target properties, previous work, strengths, and weaknesses

Technique Type	Haptic Properties	Main Strength	Main Weakness
Texture Remapping and Deformation	Compliance [4, 7]	Promising results with the usage of mainly visual feedback	Most implementations are in non-VR environments due to the complexity of the remappings and deformations
	Thermal [23]		
Perspective Distortion	Weight [41, 16, 39]	This technique is very broad and may be implemented in different ways to tackle several material properties	Most implementations need an extra device that might not be standard VR equipment (such as a mouse or touchscreen)
	Compliance [31, 43, 30, 40, 49]		
	Friction [31, 37, 48, 13, 47]		
	Roughness [33, 34, 3, 46, 22, 21]		
Sound Localization & Binaural Rendering	Dependent on the technique it is combined with [2, 19, 27, 11]	Improves the effect of other techniques (visual, passive and haptic feedback) and can provide 3D environmental information	Highly dependent on the combination with other feedback, and their studies were conducted in the area of multimodal feedback rather than in pseudo-haptics
Touch Redirection & Object Remapping	Dependent on the haptic device and/or object used [20, 28, 29, 6]	Allows the perception of most haptic properties if combined with good passive haptic feedback devices	Involves the usage of extra objects and/or devices

2.4 Summary

In this chapter, we explore the utilization of haptic feedback and, more specifically, pseudo-haptics in virtual reality (VR) applications. We first discuss the concept of haptic feedback, followed by an examination of haptic devices such as controllers, touch-sensitive surfaces, and gloves.

One significant challenge hindering the widespread adoption of haptic devices is their high cost and complexity. To address this, researchers have turned their attention to pseudo-haptic techniques as viable alternatives for simulating haptic feedback. Pseudo-Haptics rely on leveraging visual feedback to create the illusion of touch within the virtual environment.

We then delve into essential haptic concepts, starting with visual haptics (also known as visuo-haptics), which encompass systems that utilize haptic devices and visual stimulation to enable users to perceive haptic properties. It can also refer to systems that simulate haptic cues solely based on visual stimuli.

Additionally, we explore the concept of active haptics, which involves computer-controlled physical interfaces that provide objective feedback by altering the properties of the devices. Active haptic devices, such as force feedback controllers, haptic gloves, and touch-sensitive surfaces, are particularly noteworthy for enhancing users' interaction with virtual objects and environments.

Furthermore, we introduce passive haptics as interfaces that offer feedback to users without the need for computer-controlled mechanisms. These interfaces simulate the sensation of various materials or surfaces. Passive haptic interfaces are often more cost-effective and widely applicable.

In the latter part of the chapter, we explore the concept of Pseudo-Haptics and its techniques. This concept started by being defined as a combination of passive haptics with visual feedback. However, it can also represent techniques that generate haptic feedback or the illusion of touch leveraging only visual and auditory feedback or a combination thereof.

Finally, we take a look into the state-of-the-art in pseudo-haptics techniques, with a particular emphasis on adapting promising technologies and algorithms for a 3D VR environment without the need for haptic devices. These techniques are further classified into perspective distortion, texture remapping and deformation, sound localization and binaural rendering, as well as touch redirection and object remapping. Throughout the chapter, we provide examples of studies and experiments conducted for each category as well as for different haptic properties.

As we observe a lack of adaptation of promising techniques used in 2D environments to 3D VR environments when it comes to the perception of the property of compliance, we decide to dedicate this study to that possibility, utilizing only visual feedback.

Chapter 3

Deformation as Pseudo-Haptic Feedback for Compliance

Upon reflecting on the information from the previous chapters, we consider that Virtual Reality applications can grow considerably from having more detailed haptic feedback. However, studies have not yet found an answer to providing this feedback to such environments in a way that is cheap and broad. With the emergence of the concept of pseudo-haptics and their explorations, we can strive to get closer to that goal. With this in mind, the aim of this dissertation becomes clear: to explore a new take on technologies and techniques in the area of Pseudo-Haptics (specifically visual-based ones) that fit in a virtual environment and use today's widespread Virtual Reality technologies.

After taking a deep look into the current state of Pseudo-Haptics and Haptic perception in Virtual Reality, we realized that many techniques that had proven their potential in 2D mouse/cursor-based applications had not been furtherly explored and adapted into a 3D environment. For this reason, we decided to focus this investigation work on the property of compliance and inspire our technique on previous studies conducted by Ban et al. [7] and Argelaguet et al. [4]. We will be adapting some of the ideas that were previously explored for 2D environments and mouse cursors into 3D and VR experiences, as well as taking the idea of applying a C/D ratio formula to the user's hands according to the compliance levels of the touched object.

After considerable discussion and taking into account the issues raised in previous chapters, we decided to invest our work only in the visual part of the technique, therefore creating a new visual-based pseudo-haptic technique for the perception of compliance in Virtual Reality. The implementation details and the development process and decisions will be further explained in this chapter.

3.1 Design Overview

After analyzing the works of Ban et al. [7] and Argelaguet et al. [4], we started by looking closely into the known techniques that could implement the several layers required to simulate the deformation of a virtual object in order to provide information on compliance. The essential layers are the deformation physics algorithm to apply to the target object, the architecture of the technical implementation of deformation, the interface that the user utilizes to interact with the target object, the way that interface is represented in a 3D environment, and the physics algorithm to apply to that interface.

3.1.1 Deformation

Starting with the physics system, we looked into the most commonly used algorithms to simulate object deformation in the area of computer graphics. From this research, we found the most interesting to explore further to be the Finite Element Method (FEM) [8], the Position-Based Dynamics (PBD) [36], and the Mass-Spring-Damper model [17].

We decided to use the Mass-Spring-Damper model, which is a straightforward and intuitive approach representing objects as a network of interconnected masses and springs. This means that each particle of the object is represented by its equations of motion, where interaction forces are applied, dampening effects are added, and the spring force is applied to attempt to make it return to its equilibrium state. The calculations behind this method are relatively simple and, for that reason, computationally efficient. It is also able to be easily tweaked by modifying the force of the springs and the dampening modulus. Thus, it is perfect to use in real-time interactive systems, but the simulations it offers are not very complex nor very accurate when compared to the previous methods.

Additionally, we explored the possibility of using the Finite Element Method, a technique used to solve partial differential equations. It represents objects as a mesh of interconnected elements, subdividing the object into smaller regions. For this reason, this algorithm provides high flexibility in manipulation and is able to handle complex materials. However, it is computationally intensive and very hard to implement, as it requires specialized software. That said, as a 3D environment rendered for Virtual Reality is already considered to be computationally intensive, and its implementation is not in check with the time frame we had to develop this prototype, we decided this technique to be inappropriate for the purposes.

The algorithm of Position-Based Dynamics was also evaluated, as it's a very tempting technique that focuses on maintaining the position constraints between the sections, nodes, or vertexes of the object, providing an efficient, stable, and simple way of simulating deformation. However, this technique is not very flexible, as the implementation of the constraints needs to change accordingly to the properties of the object. As we aim to provide a method that is able to be applied in a wide variety of applications and allows the perception of several levels of compliance, this technique has lost its appeal.

3.1.1.1 Mass-Spring-Damper for this approach

As mentioned above, the Mass-Spring-Damper model consists of interpreting an object as a conjunction of mass nodes that are interconnected by a network of springs and dampers. A diagram for a simple system based on this model can be observed in Figure 3.1.

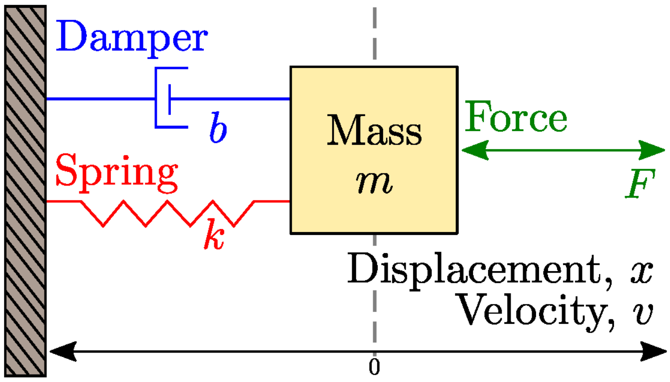


Figure 3.1: A simple mass-spring-damper (MSD) system, consisting of a mass (m) attached, simultaneously, to a linear spring with k (N/m) and a linear damper with b (N·s/m). An external force (F (N)) excites the system dynamic, which is characterized by the displacement (x (m)) and velocity (v (m/s)) of mass. (Taken from [14])

As we can see, the displacement and velocity equations of motion can characterize a system of this model. So let's start looking at the main actors in that system of motion equations.

Deforming Force

One of the main components of that system is the deforming force applied to the object. Therefore, we must start by calculating the forces applied to each mass node. The point of contact of that deforming force in the object, however, might not be a node of our network. Therefore, we need to adjust each force according to the distance from the contact point and the position of the node that we are currently analyzing. This force is called the attenuated force, and it is the main responsible for the deformation effect on the object. In our approach, we used the inverse-square law to calculate this attenuation. This means that in order to calculate the deforming force applied on a given node ($\overrightarrow{F}$n), we take the original deforming force applied on a contact point ($\overrightarrow{F}$c) and divide it by the distance between that point ($\overrightarrow{P}$c) and the current node's position($\overrightarrow{P}$n), squared. This gives us the formula:

$$\overrightarrow{F}_n = \frac{\overrightarrow{F}_c}{\left\|\overrightarrow{P}_n - \overrightarrow{P}_c\right\|^2} \tag{3.1}$$

In order to guarantee that a force has its value maxed when the distance is zero and to prevent this formula from tending toward infinity, we need to slightly modify the denominator, which gives us the final formula for the attenuated force:

$$\overrightarrow{F}_n = \frac{\overrightarrow{F}_c}{1 + \left\|\overrightarrow{P}_n - \overrightarrow{P}_c\right\|^2} \tag{3.2}$$

Now that we have the first force of our motion system, we apply Newton's Second Law to obtain the value of the acceleration that it causes to that node. In order to simplify the calculations, we will neglect the mass as if it were 1 for each node. This gives us the following:

$$\overrightarrow{F}_n = m * \overrightarrow{a}_n \Rightarrow \overrightarrow{a}_n = \frac{\overrightarrow{F}_n}{m} \Rightarrow \overrightarrow{a}_n = \overrightarrow{F}_n \tag{3.3}$$

As more deforming forces are applied (with other fingers, for example), we repeat the process to add more forces to the equation system.

Spring Force

Now that we have the main deforming force, we need to know how to calculate the other main actors of this system: the spring's resistive force, the most crucial part of the Mass-Spring-Damper model. Hooke's Law is a fundamental part of that calculation. It states that the force exerted by a spring is directly proportional to the displacement of the spring from its equilibrium position. With this law, we can calculate the spring force ($\overrightarrow{F}$s) by subtracting the equilibrium position ($\overrightarrow{P}$e) from the displaced position ($\overrightarrow{P}$d) and multiplying it by the spring force factor (Sf).

$$\overrightarrow{F}_s = (\overrightarrow{P}_d - \overrightarrow{P}_e) * Sf \tag{3.4}$$

Following this calculation, we repeat the process explained above, in which we use Newton's Second Law to acknowledge that:

$$\overrightarrow{a}_s = \overrightarrow{F}_s \tag{3.5}$$

Resultant Acceleration

Now that we have the two main acting forces in the motion system, we can calculate the resultant force and the resultant acceleration of each node ($\overrightarrow{a}$r) on each time frame. To do that, we simply add both the accelerations we calculated previously:

$$\overrightarrow{a}_r = \overrightarrow{a}_s + \overrightarrow{a}_n \tag{3.6}$$

Generally speaking, these accelerations have opposite directions, which means that, in modulus, the resulting acceleration is lower or equal to any of these accelerations. This means that it is vital to keep track of the signals when calculating the intensity of these forces and accelerations to guarantee that upon a deforming force being applied, the object deforms inwards and not outwards.

Velocity Equation

Finally, once we have the resultant acceleration, we can use the linear movement velocity equation for each node in order to obtain the velocity ($\overrightarrow{v}$r) in each time frame, which takes the starting velocity (v0) and the time frame that we are working with:

$$\vec{v}_r = \vec{v}_0 + \vec{a}_r * \Delta t \tag{3.7}$$

Damping

Moving on to the last aspect of the Mass-Spring-Damper model, we have the damping. The damping takes into account the dissipative effects of the system. This represents the resistive effects of the object's material, which helps stabilize motion and prevent excessive oscillations caused by each force that might be applied to it. While in a standard Mass-Spring-Damper system, the damping values might prove to be a complicated subject as it is a value that represents any resistances, drag, inertia, and others, in our approach, we ignored those calculations. To achieve simplicity, the damping is given its value according to the level of compliance we wished to simulate, and it takes the same value for every node of the object.

This being said, in our approach, we simply use damping as a constant factor that decreases the velocity of each vertex over time to get the final velocity ($\vec{v}$f) of each frame:

$$\vec{v}_f = \vec{v}_r * (1 - damping * \Delta t) \tag{3.8}$$

Position Equation

At last, we have everything we need to calculate the position of the deformed node. As a linear movement system can translate this, we take the starting position at the given time frame and add it to the final resultant velocity we just calculated. This being said, the position of the node can be calculated:

$$\vec{P}_n = \vec{P}_0 + \vec{v}_f * \Delta t \tag{3.9}$$

3.1.2 Interaction Interface

When it comes to the interaction with the object, in our approach, we decided to use a virtual representation of the user's hands as an interface. In order to do that, we utilized a composite of cylinders and spheres to represent the hand's bones and joints, respectively. To get an accurate representation, we employed tracking information based on hand-tracking software to retrieve the positions, rotations, and scales, as well as the connections between each part of the hand.

However, as we said previously, we utilize a Control/Display ratio-based technique. In this approach, the displacement formula is applied upon contact with the object. This means that the hand's positions stop being calculated based directly on the tracking information when they are in direct contact with the deformable object, to be instead calculated using an inverse kinematics algorithm. Further details of this interaction are described in the following subsection.

3.1.3 Object Interaction

One of the key inspirations for implementing our interaction was Newton's Third Law, which states that for every action, there is an equal opposite reaction. This means that besides deforming the deformable object, we must also apply a system that changes the user's hand and fingers to increase the realism of the technique as a whole. However, we felt that making another complex system of deforming forces for the user's fingers would not be viable. With this in mind, for the sake of our approach and as a way to increase efficiency, we decided to define the main points of interaction as the fingertips. This also makes it easier to apply an algorithm based on Control/Display. The reason why we decided to apply such an algorithm is that we felt like using unmodified tracking information at all times could prove to be very unrealistic and potentially completely break the user's immersion.

We investigated the possibilities of making the interaction between the fingers and the deformable object more realistic and immersive. Some explorations included the possibility of modifying the interaction of the object itself rather than focusing on the user's hands. However, those neither seemed appropriate to handle deforming objects nor showed the possibility of distorting the display of the interface (the hands) when compared to the control.

So the algorithm we decided to explore to achieve this purpose was based on inverse kinematics, as it is a widely used physics model in computer graphics and articulated structure simulations (which includes fingers). Additionally, we also looked into the possibility of implementing an Articulated Body Dynamics algorithm. It is similar to IK and it appears to be more realistic and plausible as it adds the possibility of implementing multiple detailed constraints to increase accuracy. However, it also has a high increase in implementation complexity and a considerable decrease in efficiency. Therefore, we considered inverse kinematics to be the most suitable for this approach.

3.2 Implementation

We are now going to discuss the implementation process and details of all the layers in our approach. We will start by explaining the implementation of the deformation algorithm, the implementation of the user's fingers as the interaction interface, the inverse kinematics algorithm used, and the interaction itself, as well as all the technologies used. The entire solution was implemented using Unity 3D and the scripting language of C#. In a 3D virtual environment, objects are represented by a Mesh. A Mesh, including in Unity, has several properties, such as vertices, triangles, and normals, among others.

3.2.1 Deformation

As we mentioned in the previous section, we are implementing an adaptation of the Mass-Spring-Damper model. With that, we considered that each of the mesh's vertices represents a mass node with a respective spring and damper. This means the algorithm's accuracy highly depends on the number of vertices of the mesh, achieving the best results when there is a considerable amount of them.

We looked into potential Unity-based architectures and possibilities to implement this model in this fashion. Since there are several ways to alter Unity meshes, and considering that the Mass-Spring-Damper model is a computationally efficient algorithm, we prioritized finding the most straightforward approach in terms of implementation rather than focusing on efficiency.

That being said, we decided to use a single-threaded implementation. We discarded the possibility of using shader-based techniques as it would require utilizing more than one programming language and thus complicate the implementation. As for the remaining options, both a C# jobs system and the usage of the MeshData class used multi-threaded implementations, which, for the sake of simplicity, we stayed away from.

While we recognize that, in terms of efficiency, this is not the best choice, it is also by far the one with the most straightforward implementation. It thus facilitates its tweaking, debugging, and adaptation to the problem at hand. This did lead, however, to a few limitations when the complete technique came to life.

As we are doing the deformation calculations in every frame and for every vertex, the efficiency decreases significantly. Besides doing the recalculation of each vertex's positions, we also need to request Unity to recalculate most Mesh properties, as it has been altered, which is a computationally intensive task. This does, anyhow, allow for the Mesh's textures, shaders, lighting, and colliders to be recalculated and adapted to the deformation, which dramatically increases the level of realism.

Nonetheless, this choice had repercussions, as the usage of an object with more than approximately 7000 vertices considerably decreased the frame rate and the overall experience, as well as making it unfeasible to calculate contact forces for multiple points of the finger, as well as for both hands. This is why, in order to simplify the implementation as well as the mathematical system, we considered that all mass nodes of the object have the same properties (equal and constant spring force factor, equal and constant damping value, and mass equal to one), which means that we can apply the same simple system of motion equations for each vertex. Additionally, in our evaluation prototype, we used a high-poly cube with approximately 1500 vertices (Figure 3.2) in order to provide high frame rates in the experience while maintaining a high level of accuracy.

As for the deformation implementation itself, starting with the interaction force, as we saw in the formulas presented in the previous section, the variables required are the deforming force applied by the user's finger, the contact point of that force, and the equilibrium position. The first two depend on the user's input, and the detailed calculations will be explained in the interaction subsection below (3.2.3). As for the equilibrium position, it needs to be stored as the starting position of each vertex of the deformable object (this is, the position in which the object loads them before any force is applied).

When it comes to the spring force, we simply require the vertex's current position (the potentially "displaced position"), the equilibrium position, and the spring force factor. For that matter,

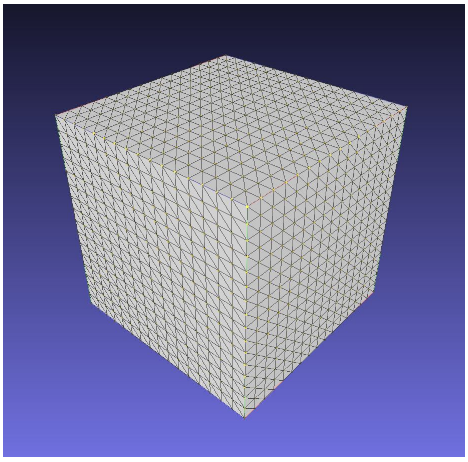


Figure 3.2: Wireframe of the high vertex cube used.

we use the current position of the vertex on each frame, and we already have the equilibrium position stored. Regarding the spring force factor, it is chosen according to the level of compliance that we wish to simulate and it is the same for every vertex of the object for the reasons we stated earlier.

Moving on to the velocity formula, as we are working with a real-time virtual environment based on Unity, everything is calculated in every frame, which means that V0 is the velocity calculated on the previous frame. The time frame is the time between each calculation (between executions of the *FixedUpdate* function), and its value can be accessed via *Time.deltaTime*. As for the resulting acceleration, we simply add the modulus of each force and add them (while taking into account the signals representative of the forces' directions). We then decrease the velocity using the formula that utilizes the damping. However, we noticed that once we changed the scale of the object (to test if the size of the object affects the perception of the user, as will be explained in the next chapter) that the physics would change significantly. To fix this issue, we divided the velocity calculated by the *localScale* used in Unity. Finally, we update the position of the vertex using the position formula.

By applying this whole process iteratively on every object vertex in every frame, we achieve a satisfactory level for the deformation simulation (Fig. 3.3). We also explored the possibility of changing how the vertex list is iterated. For that purpose, inspired by the theory of the algorithm of FEM, we defined the possibility of dividing the object into n subgroups based on the distance between each vertex and contact point. This added the possibility of changing how the deformation would be processed, as well as preventing the farthest vertices from being affected by the deforming force, which is especially useful when the object is larger and more complex. These functionalities were not explored further as we focused on achieving a general proof-of-concept rather than a complex deformation algorithm. This shows, nonetheless, that this algorithm has very high adaptability and applicability.

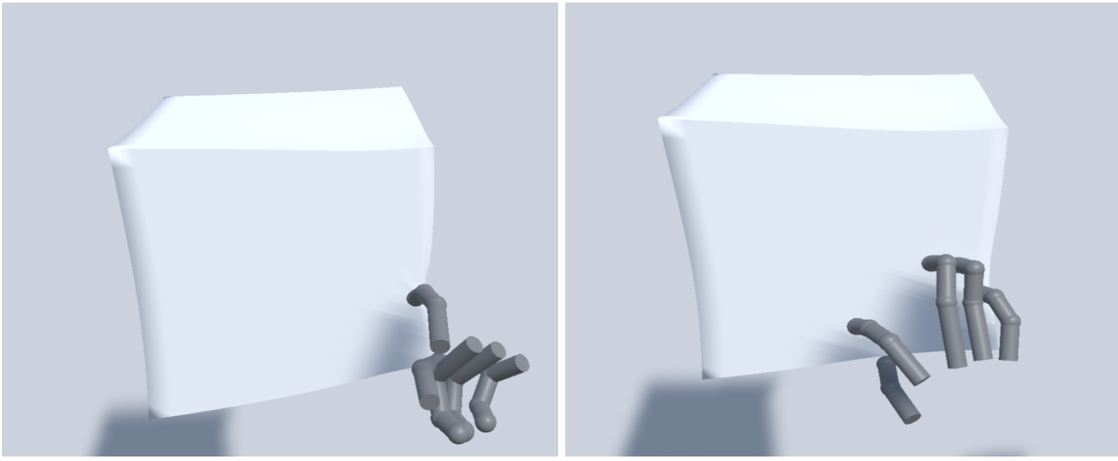


Figure 3.3: Screenshots of the deformation of the cube.

3.2.2 Interaction Interface

Regarding hand-tracking software, the main options considered were: VR Controllers, HMD Hand Tracking (provided by the HTC Vive Pro 2), and LeapMotion controller hand tracking. In order to make this choice, we developed a testing environment in which we tested the three different interfaces mentioned. We ended up discussing the possibility of using extra hardware. After some investigation and looking at the survey by Tong et al. [44], we took an interest in the LeapMotion controller, which is the most used hand-tracking hardware and, thus, the most widespread and available to the average user. The software and documentation provided by UltraLeap also considerably facilitated the manipulation and adaptation of the tracking information required to improve the level of interaction and realism of our solution.

We also tried using the VR Controllers (VIVE controllers) and investigated the study by Ganias et al. [18]. We used the Steam VR plug-in for Unity, which includes a 3D model of a hand with gloves. Even though this provided very accurate tracking of the user's hand position, the interaction was not very immersive nor realistic, as you had to be holding the controller, and it did not simulate the touching or squeezing of a deformable object at all. Besides, the manipulation of the interaction using the Steam VR plug-in is limited and could sometimes become complicated.

Additionally, we investigated the possibility of using the built-in hand tracking provided by the HMD (in this case, the HTC Vive Pro 2). At the time of this study, the XR hands plug-in provided by Unity to handle the VIVE tracking (asset shown in Figure 3.4) was still in beta. Therefore it showed several issues and low accuracy on the tracking. However, it is a very attractive solution to provide tracking using just the HMD. Consequently, it could be explored in the future using an adapted version of the solution we are presenting.

We evaluated the hand rig provided by UltraLeap itself, which was by far the most accurate when it comes to hand tracking. It proved to be very efficient, even though it was re-calculating the positions and redrawing the hand model based on a precise tracking of the user's hands (given by the CapsuleHands prefab, which is represented in Figure 3.5). However, even though we tried,

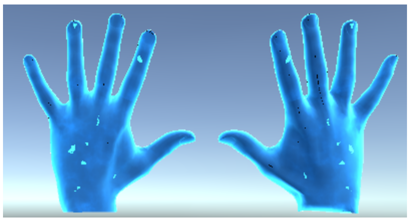


Figure 3.4: The hands rendered using the XR hands plug-in and the VIVE tracking.

we were not able to use this piece of software, as it did not allow us to manipulate it freely when it came to its behavior upon interacting with the deformable.

As for the implementation itself, to use the LeapMotion controller, we had to install the latest version of UltraLeap's tracking software, named Gemini. Additionally, we also had to install the required Unity plug-ins, as explained in the "Getting Started" part of UltraLeap's documentation for hand tracking in Unity.

For this reason, we decided to draw our own hand model, using Unity's cylinders (to represent the bones) and spheres (to represent the joints and the tip). We linked that to the information provided by the LeapMotion controller. In order to do that, we had to iterate through the Hand object's Finger instances (provided by UltraLeap's software) and use the members' (Bone instances) variables to determine the positions, rotations, and scales of both the joints and the bones. This made the implementation significantly less efficient, but it simplified the implementation and increased the freedom of manipulation, which became crucial to improve and implementing the interaction that will be described below.

This process requires, however, our object to be updated on every frame based on the tracking information, which is also a computationally intensive task. This hand object will be the representation of the "Control" part of the Control/Display ratio-based technique we will be implementing for the interaction. This means that it is always representative of the user's real hand and will be used to calculate the intensity and direction of the forces used in the deformation algorithm used.

As is standard in a Control/Display ratio-based technique, the Control values are not the ones shown on the screen. In our implementation, the hand object we just described is also not the one displayed to the user. Instead, we created a similar object but in a different structure (which was required for the inverse kinematics asset) that is representative of the "Display" hand. This new

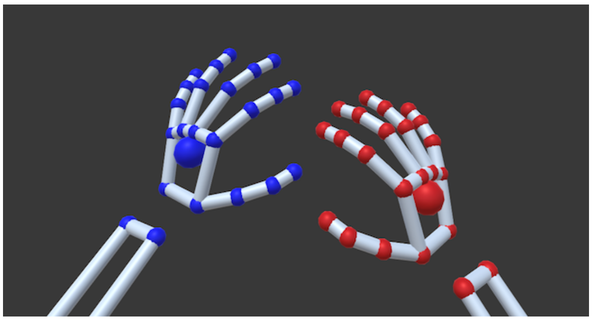


Figure 3.5: The representation of the CapsuleHands prefab provided by UltraLeap.

object is responsible for both the virtual hand that the user can observe in the virtual environment and the calculation of the contact points of the interaction. In order to achieve this, we developed a script that links the properties of each component of the Display object to match the respective Control ones. This script can be easily toggled on or off so that we can manipulate the Display object freely when required upon contact with the deformable.

As a way to increase efficiency, we decided to stop rendering and processing parts of the hand that we considered to be irrelevant to the perception of the levels of compliance. This means that our hand representations lack their palms, meaning that they only include the representation of the three finger bones (excluding the thumb, which only has two) and the respective joints and tips. The displayed version of the hand in our prototype is shown in Figure 3.6.

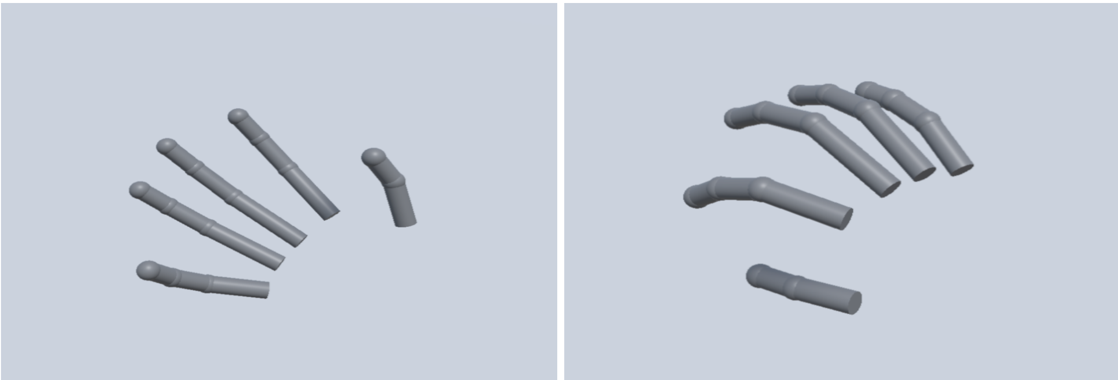


Figure 3.6: Screenshots of the user's displayed hand in our approach.

3.2.3 Object Interaction

3.2.3.1 Inverse Kinematics and Control/Display

The final part of our implementation includes how the interface interacts with the object and how the interactions are calculated. Starting with the interface interaction with the object, we investigated different types of inverse kinematics algorithms. We decided to use an implementation of FABRIK [5](Forward and Backward Reaching Inverse Kinematics). We also looked into other algorithms that are currently commonly used in the area of computer graphics, such as the Jacobian Transpose Method [15], Cycle Coordinate Descent (CCD) [26], and Damped Least Squares (DLS) [12]. Out of those algorithms, only two of them had simple implementations available in the Unity Store as free assets: CCD and FABRIK. The implementation of CCD we found in that asset was only able to accurately calculate the simulation of an entity with two bones and one joint. For this reason, it was unable to provide a satisfiable simulation of a human finger, which (besides the thumb) has two joints and three bones.

The other free asset available was called "Fast IK" ([1]) and used the FABRIK algorithm. This algorithm proved to be perfect for usage in a chain such as a human finger, as it is able to handle

[1] https://assetstore.unity.com/packages/tools/animation/fast-ik-139972 (Last accessed: 21/06/2023)

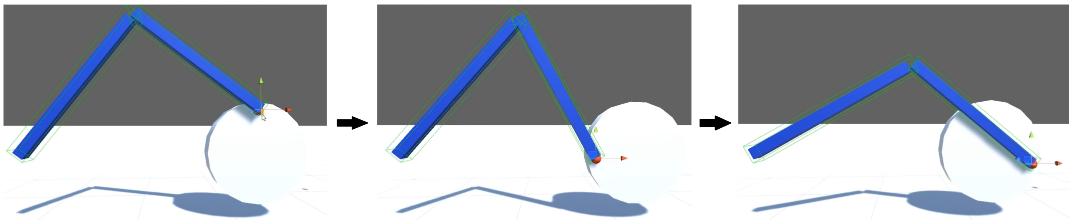


Figure 3.7: The FastIK asset's calculations along a circular movement of the target - the red sphere.

more complex kinematic structures in an efficient way. Figure 3.7 represents the results of this algorithm in a basic setting where the kinematic chain follows a red sphere that has a circular movement.

This algorithm operates by iteratively updating the joint positions along the fingers, starting from the distal end and progressing towards the chain's root (forward reaching) and subsequently backtracking towards the fingers' distal end (backward reaching). This iterative process continues until the target position is reached or when it is sufficiently close to it.

In our implementation, we applied this algorithm to the Display object upon contact with a deformable object after disabling the script that links it to the Control object. As the point of that contact is always at the fingertips, it also made it easier to implement an accurate simulation of this interaction.

To illustrate the effect created by the Control/Display technique that we described, we modified the rendering options of our prototype to take some representative screenshots. This can be seen in Figure 3.8.

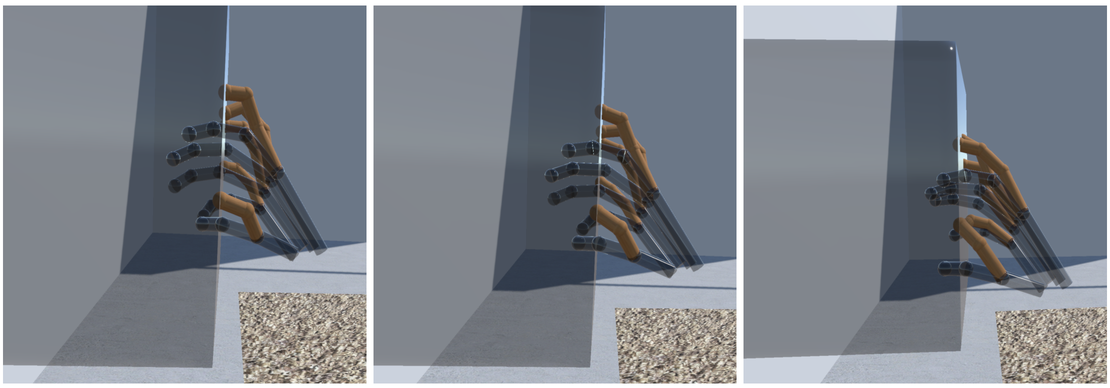


Figure 3.8: The Control hand is rendered in glass material, while the Display hand has a wooden-like material. When the user starts applying force to an object, we use the Control hand to calculate the intensity of the deforming force, while only displaying the Display hand that is calculated by the FABRIK algorithm.

After testing the asset for an interactive real-time application such as ours, we realized that it sometimes provided inaccurate results due to the constant alteration of both the user's finger's positions and the target position post-deformation. Therefore, we investigated the possibility of adding constraints to the algorithm. Still, we ended up deciding against it because of the complexity that comes with altering an official asset from the Unity Store, as well as a lack of time to explore the algorithm in detail and its implementation to make those modifications.

3.2.3.2 Deforming force

Getting into further technical detail when it comes to the deforming force, when each of the user's fingertips collides with the deformable mesh, the two parts of the interaction are triggered: the inverse kinematics algorithm activates with the target position being the collision point, and a deforming force is added to the motion system of the deformable mesh.

A vital part of this interaction is how the forces will be applied to the deforming mesh and how they will relate to the user's input. The first part of this process is calculating the contact point between a user's fingertip and the deforming mesh. In order to do this, we first tried to utilize the functionalities of Unity's Mesh Collider components directly, applied to both the tips and the mesh, to calculate collisions and their properties. However, several issues arose during that experiment.

The first problem was that that component in the version of Unity used for the prototype did not support collisions that included colliders of meshes with over 256 vertices without calculating a convex approximation. Furthermore, another issue was that, when using Unity's collider components, the collisions apply the Unity physics automatically unless they are set up as being "Trigger" colliders. Given all of this information, we added a "Trigger" Mesh Collider to the deformable mesh and no collider to the fingertips. We instead iterated through the user's fingers on each frame to check if their tips were inside the deformable mesh's collider bounds. Besides we also stored the position of each tip to be interpolated with the one in the next frame. If any of the fingertips are inside the bounds, we then calculated an interpolated direction of the movement using the stored information. Given this direction, we cast a ray to get an accurate pinpoint of the contact point with the mesh.

Additionally, we utilize that condition (if a fingertip is within the collider's bounds) to toggle the calculation of the Display object using the inverse kinematics algorithm described above. We apply that algorithm to the finger that is in contact with the deformable, thus toggling off the linking script for the components of that finger while keeping the remaining hand components controlled by the tracking information. Once the fingertip stops being in contact with the object, we toggle on the linking script again to have the Display equaled to the Control.

After having the contact point, we need to calculate the force's intensity. To achieve that purpose, the Control object really comes into play. Even though the hand being rendered during contact is calculated by the inverse kinematics algorithm, we can still use the Control object, which is linked to the tracking information, to calculate, in each frame, the magnitude of the direction vector. We then use that value to increase the force intensity linearly. However, as the force is cumulative in each frame, we had to define a maximum intensity for each finger. This process is repeated every frame for each of the user's fingers, which allows for multiple forces to be added to the system (one for each finger).

Finally, we have all the information required to apply a deforming force to the object. This is done by accessing the object associated with the collider that we are interacting with and adding a force to its Mass-Spring-Damper force system, described above, which will simulate the deformation caused by that interaction and, according to the parameters we chose, give feedback regarding the compliance of the object.

3.3 Summary

This chapter explores a new approach to pseudo-haptics in virtual reality (VR) environments, aiming to enhance haptic feedback in a cost-effective way. The proposed technique focuses on three main layers: the deformation algorithm, the interaction interface, and the object interaction.

The deformation algorithm is based on the Mass-Spring-Damper model. We start by referring to its advantages over other methods, followed by a detailed explanation of how it works in our approach.

Regarding the interaction interface, we decided to use a virtual representation of the user's hands, based on cylinders and spheres as bones and joints. To do that, we use hand-tracking information. However, when the hands come into contact with an object, we instead use inverse kinematics.

As for the object interaction, we use the tracking information to receive the input required to calculate the deforming force and utilize inverse kinematics as the reaction to that force on the user's fingers. Other options besides inverse kinematics were explored, but for efficiency reasons, we decided it was the most suitable.

Moving on to the implementation, the entire process of implementing the deformation system using Unity 3D's meshes is documented, including technical decisions and details. The implementation of the interface was brought by UltraLeap's LeapMotion controllers, which provided the hand-tracking information required for the Control and Display objects to be constructed. At last, we outline how the inverse kinematics algorithm of FABRIK works in our approach, as well as the reasoning behind that choice. The chapter ends by describing our the details of the deforming force are calculated in the implementation and how it is added to the deforming system.

Chapter 4

Evaluation and Results

An effective pseudo-haptic technique must enhance the user's immersion and perception of a given property, increasing the realism of the virtual environment. Therefore, testing their abilities with real users is mandatory to evaluate their impact on their experience better.

This technique was conceived to identify the level of compliance of a virtual object, so we intend to evaluate the relative perception of that property in a pair of virtual objects. To better evaluate the users' experience, the requested interaction is simple, but the users are free to touch each virtual object as it best suits them.

In this evaluation, we aim to assess if, with this technique:

- the users can successfully identify different levels of compliance for visually equal virtual objects;
- the virtual object's size affects the perception of compliance;
- the visual characteristics (textures) affect the perception of compliance;
- a realistic and immersive way of perceiving virtual object compliance can be achieved.

4.1 Prototype

With the aim of testing our technique, we set up a virtual environment developed with the Unity Engine and the C# programming language. This virtual environment consisted of a well-lit room with a testing area, that included a TV screen that displayed information regarding the trial that the participant was on and 2 High-poly Cubes, whose properties would vary according to the task and trial. Additionally, a LeapMotion Controller was used to display one of the user's hands virtually. More information regarding the implementation details of the prototype is presented in the previous chapter.

4.2 Setup

The users wore an HTC Vive Pro 2 headset to visualize the virtual environment that was set up for them, that had a LeapMotion Controller glued to it on the camera area, in order to override the hand tracking system and allow the users to see one of their hands while interacting with the cubes. The SteamVR bases were placed at an approximate height of 2.20m and with a distance of approximately 4m. The setup can be seen in Figure 4.1. Figures 4.2 and 4.3 show participants participating in the tasks.

Figure 4.1: HTC Vive headset and the LeapMotion Controller attached.

Figure 4.2: A participant doing the tasks.

Figure 4.3: Another participant doing the tasks.

4.3 Participants

A total of 26 participants took part in the experiments, with all of them resulting in valid responses. Of these 26 participants, 20 are males and 6 are females, aged between 14 and 67 years old. Half of the participants had never used VR before, 11 had used it once or twice, 1 used it annually, and 1 used it daily. Most participants (21) are currently students, with 16 of them being students of Informatics Engineering. As for habilitations, almost all participants had bachelor's degrees (22). The plots showing the distributions of these stats can be consulted in Appendix A.1.

4.4 Procedure

The tests took place at FEUP, in room I003, which is a laboratory dedicated to students studying for their Master's Degree in Informatics Engineering, and it provided a zone with the necessary space for the experiments. To thank the users for their participation, they were rewarded with a chocolate bar at the end of the experience.

The users were, asked to start by reading and filling in some information, and were then guided to the dedicated zone mentioned above. Before starting the virtual environment tasks, the participants had to fill in a form that stated their age, gender, area of studies, habilitations,

Table 4.1: Combinations for Each Task

Task Combination	Compliance Levels	
	Cube A	Cube B
Combination UU	U	U
Combination SU	S	U
Combination AU	A	U
Combination VU	V	U
Combination US	U	S
Combination SS	S	S
Combination AS	A	S
Combination VS	V	S
Combination UA	U	A
Combination SA	S	A
Combination AA	A	A
Combination VA	V	A
Combination UV	U	V
Combination SV	S	V
Combination AV	A	V
Combination VV	V	V

profession, and experience with VR. They were also asked to read important information regarding the experiment and give their consent to share their test data.

They would then equip the VR headset and make sure that the virtual environment was correctly displayed, centered and clearly visible. To increase the comfort of the participants, it was suggested that they would remain seated for the duration of the experiment.

The users would then be instructed to perform the tasks described in the next section and inquired about their answers. Their answers would be written down by a moderator in a form for further analysis.

The virtual experiment was expected not to take longer than 25 minutes.

After the users completed all of the trials, they would then be asked to fill in a simple satisfaction form, as is described in the next section.

The forms used can be consulted in the Appendix A.2.

4.5 Tasks

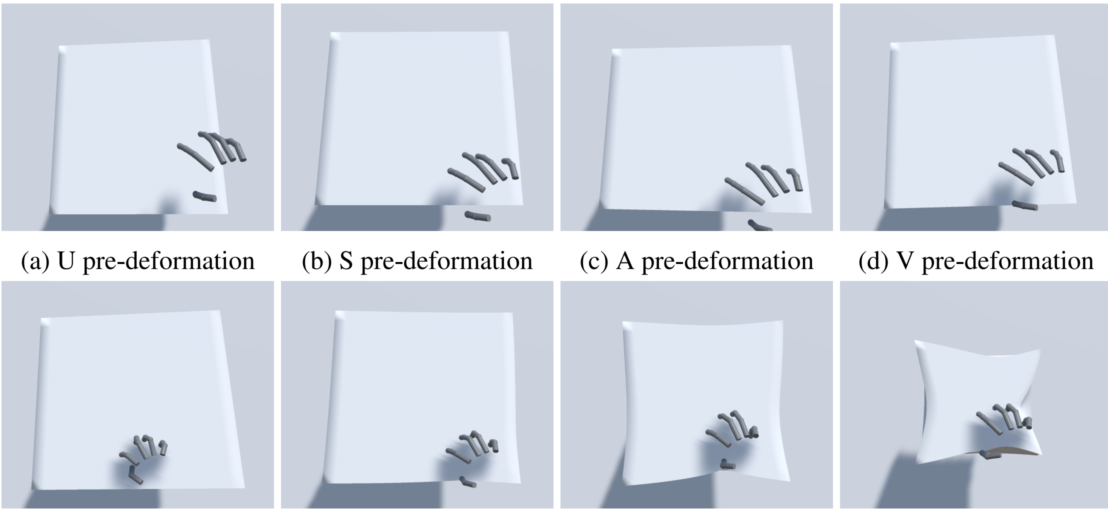
The experiment comprised three main tasks, where in each, the user is presented with a pair of cubes. For each task, there are 4 different levels of compliance for each object in the pair, making up 16 combinations. For each of the different combinations, the user is asked to identify which of the cubes had higher levels of compliance when compared to the other or if they were equal.

The levels of compliance that we decided to test were defined as "Uncompliant" (U), "Slightly Compliant" (S), "Average Compliant" (A) and "Very Compliant" (V). The combinations of levels of compliance for each task can be observed in Table 4.1.

Table 4.2: Compliance Levels and Parameters

Compliance Level	**Bounce-Back Force (Spring Force)**	**Resistance (Damping)**	**Affected Volume Ratio**
Uncompliant (U)	400	100	1
Slightly Compliant (S)	400	83	1
Average Compliant (A)	190	37	1
Very Compliant (V)	60	40	1

Each of these compliances used different parameters in the algorithm. These parameters were chosen empirically. The combination of those parameters can be seen in Table 4.2. A representation of an average deformation for each of these compliance levels can be seen in Figure 4.4



(a) U pre-deformation (b) S pre-deformation (c) A pre-deformation (d) V pre-deformation

(e) U during deformation (f) S during deformation (g) A during deformation (h) V during deformation

Figure 4.4: Representation of deformation of each level of compliance.

To check the participant's confidence level in their answer, besides being questioned on which cube they considered to have a higher level of compliance, they were also asked to give a confidence level from 1 (not confident at all) to 5 (very confident and certain). In order to even out the distribution of combinations, we used a random trial selector, since a Latin Square distribution was not suitable because there was a very large number of combinations and a low number of participants.

In the **first main task (Task 1)** - Fig. 4.5 -, the cubes presented are visually equal, and only the levels of compliance would vary. In the **second main task (Task 2)** - Fig. 4.6 -, besides varying the levels of compliance, the cubes were also presented in different sizes, where one is larger than the other (Cube A is larger than Cube B). For the **third and final main task (Task 3)** - Fig. 4.7 -, we would vary the levels of compliance in cubes that are equal in size but that have different textures (one of the cubes - Cube A - has a texture of a compliant material - a sponge -, while the other one - Cube B - has the texture of an uncompliant material - wood).

After completing all of the virtual environment tasks, the users were also asked to fill in a

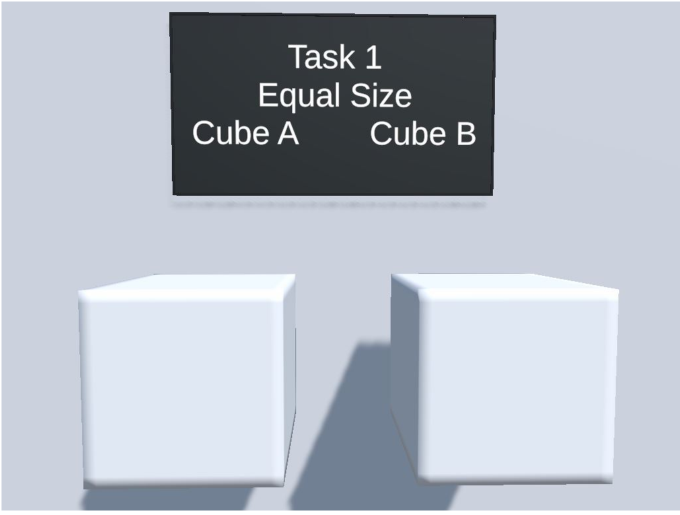


Figure 4.5: Screenshot of Task 1's setup.

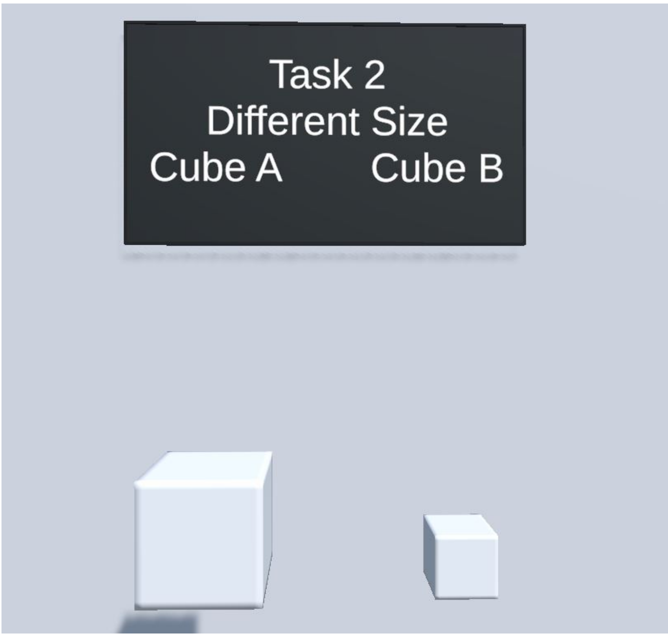


Figure 4.6: Screenshot of Task 2's setup.

form to evaluate the level of realism provided by the deformation technique alone, by the basic Inverse Kinematics algorithm as a Control/Display ratio tool, and the combination of both. In total, participants completed 1248 trials (3 tasks x 16 combinations x 26 participants).

4.6 Results

When it comes to the study of the outcome of the conducted user experiments outlined previously, we strive to comprehensively analyze both objective and subjective data we collected during the experimentation procedure described.

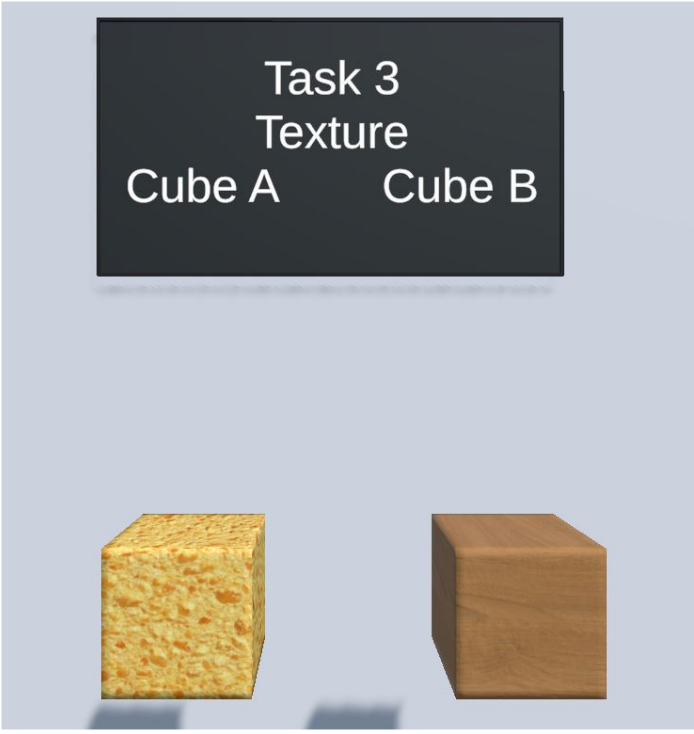


Figure 4.7: Screenshot of Task 3's setup.

The main focus of this study was the examination of the proposed technique's effectiveness in providing accurate information regarding the compliance level of an object, as well as an overall perception of a user's satisfaction level with the proposed solution. For that reason, we will examine metrics such as success rate and confidence level alongside subjective metrics, which consist of user feedback regarding realism, immersiveness, potential and overall experience.

In order to analyze the results objectively, we conducted statistical tests on both objective and subjective metrics. This allows us to draw conclusions from the gathered responses easily. On pertinent tests, we used the conventional alpha value of **5%** ($\alpha = 0.05$) to determine if there is a statistically significant difference between the tested pairs.

4.6.1 Task Results

Regarding the task responses, we started by doing a preliminary analysis where we utilized success rate percentage, average, median, and interquartile ranges. We also looked into the possibility of identifying and eliminating outliers, but we considered that they could represent relevant information. This analysis can be observed in Table 4.3.

Table 4.3: Success rate and levels of confidence for each trial of each task

Task		Trial																Avg
		UU	SU	AU	VU	US	SS	AS	VS	UA	SA	AA	VA	UV	SV	AV	VV	
1	**Success Rate - %**	96.2	92.3	100	100	84.6	76.9	100	100	100	100	57.7	92.3	100	100	88.5	73.1	91.3
	Confidence - Median (IQR)	5 (0)	5 (0)	5 (0)	5 (0)	5 (0)	4 (0.75)	5 (0)	5 (0)	5 (0)	5 (0)	4 (1)	5 (0)	5 (0)	5 (0)	5 (0)	4 (2)	4.7
2	**Success Rate - %**	100	96.2	100	100	100	80.8	100	57.7	100	100	65.4	57.7	100	100	73.1	76.9	88
	Confidence - Median (IQR)	5 (0)	5 (0)	5 (0)	5 (0)	5 (0)	4 (1.5)	5 (0)	4 (2)	5 (0)	5 (0)	4 (1.75)	4 (2)	5 (0)	5 (0)	4 (1)	4 (2)	4.6
3	**Success Rate - %**	100	100	100	100	100	100	100	96.2	100	100	88.5	92.3	100	100	88.5	88.5	97.1
	Confidence - Median (IQR)	5 (0)	5 (0)	5 (0)	5 (0)	5 (0)	4 (1.75)	5 (0)	5 (0)	5 (0)	5 (0)	4 (1)	5 (1)	5 (0)	5 (0)	5 (0.75)	4 (1)	4.8

By looking at this table, it is visible that the average accuracy on every task was high. This indicates that our technique was indeed able to provide accurate information regarding the compliance of an object. Besides, both the average level of confidence and the medians were also very high (4.6 and equal to or above 4, respectively), similarly indicating that our solution gives precise feedback.

4.6.1.1 Success Rate

Given the first research question we presented at the start of this chapter, and considering that all participants declared that they were experiencing something like this for the first time, achieving an average success rate of approximately 91% in the first task, and the lowest accuracy in a trial being close to 60% are very positive signals. Additionally, this lower accuracy can be explained based on some of the feedback that we got from the participants in the first task, where a large group told the experiment moderators they kept trying to identify which cube was the most compliant, thus forgetting the possibility of them being equal.

As for the remaining assessment questions, we used the first task results as our base for comparison since it is the one that has no variation of variables. Considering this, we employed McNemar's test between the responses in each trial (whether they were successful or not) between Task 1 and Task 2 and between Task 1 and Task 3. As some trials had no response variation whatsoever between the relevant tasks, we skipped them.

The output table comparing **Tasks 1** and **2** is available in Table 4.4. From this table, we can observe that from Task 1 to Task 2, there was a statistically significant difference in the accuracy of trials **VS** and **VA** with p-values of **0.001** and **0.022**, respectively. Based on table 4.1, if we take a look into these trials, Cube A has the maximum compliance level in both, while Cube B is slightly compliant and averagely compliant, respectively. However, in Task 2, since Cube A is considerably larger than Cube B, the participants (as we can see by looking at the success rates for these trials in Table 4.3) saw a considerable decrease in their success in the identification of the compliance levels.

As for the comparison between **Tasks 1** and **3**, it's available in Table 4.5. Taking a look at this table, we can observe that from Task 1 to Task 3, there was a statistically significant difference in the accuracy of trials **SS** and **AA**, with p-values of **0.031** and **0.008**, respectively. Again, by looking into table 4.1 and analyzing these trials, we can see that they evaluate Cubes of equal compliance but with variation in their textures. Looking into the success rates, we can see that the participants saw a significant increase in success in identifying equal compliance levels. This variation could be explained by two different factors: as the cubes in Task 3 have different but more realistic textures it makes it easier for users to perceive when the compliance levels are the same; or, since this was the final task, the users were far more familiar with the prototype and the technique itself, and learned how to improve their perception of equal levels of compliance. If we consider the latest, then that also might have had an influence on the results of the comparison between Task 1 and Task 2.

Table 4.4: McNemar test between the corresponding trials of Tasks 1 and 2 for success rate.

	T1UU & T2UU	**T1SU & T2SU**	**T1US & T2US**	**T1SS & T2SS**	**T1VS & T2VS**	**T1AA & T2AA**	**T1VA & T2VA**	**T1AV & T2AV**	**T1VV & T2VV**
N	26	26	26	26	26	26	26	26	26
Exact Sig. (2-tailed)	1.000	1.000	0.125	1.000	0.001	0.727	0.022	0.289	1.000

Table 4.5: McNemar test between the corresponding trials of Tasks 1 and 3 for success rate.

	T1UU & T3UU	**T1SU & T3SU**	**T1US & T3US**	**T1SS & T3SS**	**T1AS & T3AS**	**T1AA & T3AA**	**T1VA & T3VA**	**T1AV & T3AV**	**T1VV & T3VV**
N	26	26	26	26	26	26	26	26	26
Exact Sig. (2-tailed)	1.000	0.500	0.125	0.031	1.000	0.008	1.000	1.000	0.289

Table 4.6: Wilcoxon Signed-Ranks test between the corresponding trials of Tasks 1 and 2 for the level of confidence

Trial	Asymp. Sig (2-tailed)
T2UU - T1UU	0,705
T2SU - T1SU	0,783
T2AU - T1AU	1,000
T2VU - T1VU	1,000
T2US - T1US	0,157
T2SS - T1SS	0,819
T2AS - T1AS	1,000
T2VS - T1VS	0,002
T2UA - T1UA	1,000
T2SA - T1SA	1,000
T2AA - T1AA	0,858
T2VA - T1VA	0,006
T2UV - T1UV	1,000
T2SV - T1SV	0,317
T2AV - T1AV	0,001
T2VV - T1VV	0,815

For this reason, we would also like to mention the trial of **AV** in Tasks 1 and 2, as it sees a decrease in the success rate of over 15%, which might indicate that the variation in the object's scales influenced its results. This will be supported by the analysis in the following subsection on the variation of the user's confidence levels.

4.6.1.2 Level of Confidence

On this note, we utilized the Wilcoxon Signed-Rank test to compare the user's confidence level variation from Task 1 to Task 2, as well as from Task 1 to Task 3 in each trial.

The output table for that test comparing **Tasks 1** and **2** can be observed in Table 4.6. By analyzing this table, we can notice that there is a statistically significant difference in the confidence levels presented in trials **VS**, **VA**, and **AV**, with p-values of **0.002**, **0.006**, and **0.001**, respectively. This coincides with the above analysis regarding the success rates, which means that it is very likely that the object scale affects the perception of compliance, especially in higher levels of compliance with slight differences.

As for the output for that test comparing **Tasks 1** and **3**, it's shown in Table 4.7. If we take a close look at this table, we can see that there is no statistically significant difference in any trials. This means that it is unlikely for the texture to have an influence on the user's perception of compliance on any trial. This further supports the theory that the users might have simply developed their perception skills and familiarity with the prototype during the previous tasks.

Table 4.7: Wilcoxon Signed-Ranks test between the corresponding trials of Tasks 1 and 3 for the level of confidence

Trial	Asymp. Sig (2-tailed)
T3UU - T1UU	0,480
T3SU - T1SU	0,157
T3AU - T1AU	1,000
T3VU - T1VU	1,000
T3US - T1US	0,157
T3SS - T1SS	0,260
T3AS - T1AS	1,000
T3VS - T1VS	0,792
T3UA - T1UA	1,000
T3SA - T1SA	1,000
T3AA - T1AA	0,053
T3VA - T1VA	0,483
T3UV - T1UV	1,000
T3SV - T1SV	1,000
T3AV - T1AV	0,096
T3VV - T1VV	0,146

4.6.1.3 Analysis

After taking a close look at our results, we can confidently conclude that our technique objectively allows the perception of discrepant levels of compliance in objects that are visually equal, with different sizes, and with different textures, as the combined average success of participants in all of the experimented trials surpassed 90%. The mean confidence level in all experimented trials also surpassed 4.6, meaning that not only were the participants able to distinguish the compliance levels successfully, but our approach also allowed them to do it confidently.

The main issue arose when the scale of the objects was different, and one of the compliance levels was very high (V). This means that the perception of higher levels of compliance is more subjective to scale changes. Additionally, we can conclude that, on average, a participant is more likely to identify a smaller object as more compliant than it actually is and a larger object as less compliant than it actually is.

Another conclusion we can draw from these results is that it is easy for participants to identify compliance differences when one or more stiff objects (U and S) are involved (the exception being the one mentioned above). The final aspect we can take from these results is that perceiving equal levels of compliance might have a learning curve and is consistently the most challenging task (the exception being when stiff objects are involved) for our approach. However, this seems to be combated by adding more realistic textures, as the users improved their results in the last task for these trials.

4.6.2 Subjective Results

Regarding subjective feedback, we interrogated the participants on three different topics. The specific questions are in Appendix A.2 in the Satisfaction Form section.

In the first topic, we inquired the users on feedback regarding the general experience provided by our approach, more specifically, how easy they considered the perception of compliance was (Q1.1.), and if they considered that it could increase the realism of virtual environments (Q1.2.). On the second topic, we asked for feedback on the object deformation algorithm, particularly how they rated the accuracy of the physics applied to the object (Q2.1.), along with how many different levels of compliance they thought they had identified during the entire experiment (Q2.2.). Finally, on the last and third topic, we requested feedback on the interaction interface - the fingers. There, we asked for a rating on the overall physics of the tracking and the representation of the fingers (Q3.1.) and if they considered that the deformation of the fingers upon contact with the area of interest (the application of the IK algorithm) increased their ability to perceive compliance (Q3.2.).

All of these questions except for Q2.2. had their answers vary from a rating of 1 (Strongly Disagree or the lowest level) to a rating of 5 (Strongly Agree or the highest level). Q2.2. instead allowed users to write any number that they felt was representative of the different compliance levels they perceived during the entire experiment. They were also asked to explain their thoughts to the moderator to allow for a better understanding of their answer. While filling out this form, the users were always welcome to give any additional feedback or explanation of their answers.

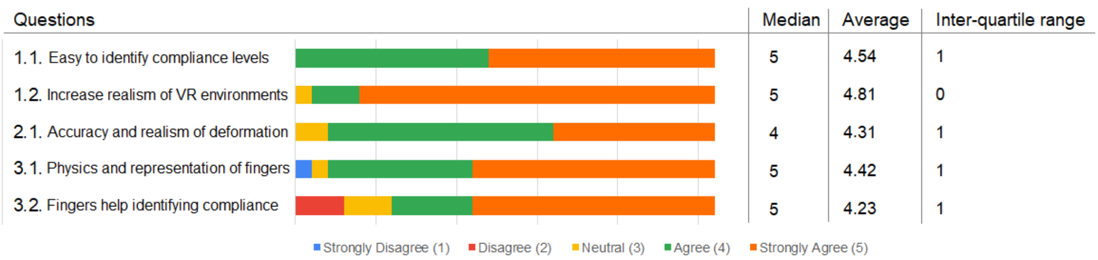


Figure 4.8: Results of the satisfaction questions.

The primary analysis for these questions, except for Q2.2., included the median, the average, and the interquartile range (Figure 4.8). Regarding Q2.2., we used a boxplot to analyze the results (Figure 4.9).

The entire distribution of answers for these questions in bar charts can be found in Appendix A.3.

4.6.2.1 Analysis

Interpreting the results of the feedback given and presented in the plots and stats mentioned above, we can see that our approach was very welcomed and praised by a majority of our participants in every aspect. With an average rating as well as medians equal or superior to 4 out of 5, we can

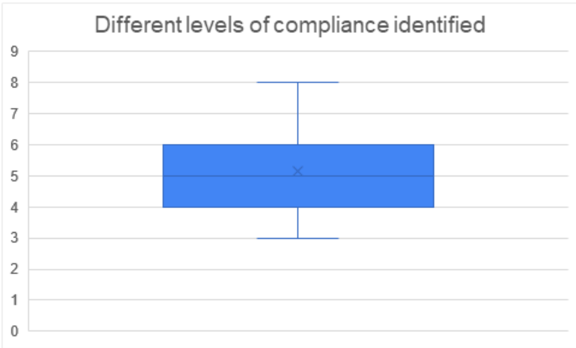


Figure 4.9: Boxplot of the perceived levels of compliance.

conclude that our participants considered our approach to be straightforward, simple, realistic, and physically accurate.

Considering the answers to Q2.2., as we can see in the boxplot (Figure 4.9, the most common responses were between 4 (which is the correct answer) and 6. During the filling of the form, we asked the participants to try to tell us to describe the different levels of compliance they were able to identify and an estimation of when they identified it. Every person that was available to explain their thought process told us that they were able to quickly identify (correctly) the uncompliant, slightly compliant, and very compliant levels. Additionally, most participants were also able to identify the average level. Still, they also affirmed that, during Task 2, they felt like there were additional levels of compliance that were close to either the average or the very compliant levels. This supports the deduction that we did in the previous section, where we stated that the scale of the object does have an influence on the perception of compliance, especially at high levels of compliance.

4.7 Discussion

Given the results and analysis presented in the Results section, we gained valuable information regarding the possibility of increasing the perception of object compliance in Virtual Reality environments. We were able to reach our goal, proving that it is possible to accurately provide feedback in the broadest of situations, including for visually equal objects, with different scales and with distinct textures.

Taking into account the results of McNemar and Wilcoxon Signed Ranks tests, we were also able to identify that the object's scale does have a slight influence on the perception of compliance, especially in situations that involve high levels of compliance. From this information, we also determined that users tend to consider smaller objects more compliant than bigger ones. Additionally, we also found that objects with lower levels of compliance (stiffer objects) were perceived much easier, as they presented accurate simulations that achieved high levels of realism.

Considering the subjective feedback given by the participants, we consider that we were able to provide a technique that is realistic and immersive enough, that provides accurate physical simulations of both a deformable object as well as of an interactive interface such as the human fingers. Besides, it is based solely on visual elements, thus having broader applicability and no requirement for additional haptic devices.

During the writing of this study, we found a work that also evaluates stiffness/compliance using pseudo-haptic techniques with many similarities to the ones used for the sake of this study. Bouzbib et al. [10] explored the possibility of providing feedback on the property of compliance using a combination of a passive haptic device with a visual approach that is close to the one presented in this study.

Bouzbib et al. [10], besides presenting an algorithm for deformation and the perception of compliance, mainly focused their work on evaluating ranges and thresholds of type of approach, especially when using a rigid tangible to provide haptic feedback. Both they and we were able to conclude that users are able to perceive a high range of levels of compliance with this approach. They also concluded that lower compliance levels (which, using our categories, include the levels of U, S, and an intermediate level between S and A) are more easily perceived and that the deformation algorithm is at its highest potential in those situations. Additionally, they also aspired to evaluate whether an object's scale has an influence on the perception of this property. Still, they were unable to collect data that fully supported that hypothesis. Nonetheless, they did find hints that the algorithm had the most success in larger objects with lower compliance levels. This further supports our results, which showed that when larger objects have higher compliance levels, the user's perception accuracy decreases when compared to smaller objects.

In conclusion, we feel like these studies complement each other and show an enticing opportunity to investigate this approach further. They tend to showcase drastic enhancements in the perception of compliance in Virtual Reality environments using a realistic and accurate pseudo-haptic technique, thus reducing costs and becoming further available to the users.

4.8 Summary

This chapter starts by talking about the conduction of user tests and the evaluation of the pseudo-haptic technique. The goal was to assess users' perception of compliance in virtual objects and determine factors affecting it, specifically size and texture. The tests involved 26 participants, including both experienced and inexperienced VR users. Participants completed trials where they compared pairs of virtual cubes in three different tasks. These presented different compliance levels and, respectively, equal size and equal texture, different size and equal texture, and equal size and different texture. The users were asked to provide insights regarding their ability to differentiate compliance, the impact of object properties, and the realism achieved, using a satisfaction form.

Subsequently, we present the results of our user experiments evaluating the effectiveness of our proposed compliance perception technique. We conducted a comprehensive analysis of both objective and subjective data collected during the experimentation phase. Our study aimed to assess the accuracy of our technique in providing compliance information and user satisfaction. We analyzed success rates, confidence levels, and user feedback on realism, immersiveness, potential, and overall experience. Statistical tests were performed using a significance level of 5%. The analysis included success rates, average accuracy, confidence levels, medians, and interquartile ranges.

The results showed high accuracy and confidence in compliance assessments, especially for objects of equal scale. However, scale variations influenced perception in a couple of trials, with smaller objects appearing more compliant. As for subjective feedback, participants found our approach straightforward, realistic, and physically accurate, correctly identifying compliance levels and acknowledging the impact of object scale on perception.

Overall, our study demonstrates the potential of our technique to enhance compliance perception in Virtual Reality, supporting our hypothesis and aligning with related research. Participants reported high confidence in their compliance assessments, and subjective feedback praised the approach's simplicity and realism. The findings highlight the influence of object scale on compliance perception and the benefits of adding realistic textures.

Chapter 5

Conclusions

Virtual Reality has been a target of many investments and explorations in recent years as it provides users with immersive environments, thus having the potential to improve professional and entertainment experiences. However, haptic feedback still has much room to improve. With this in mind, we decided to explore the concept of Pseudo-Haptics, particularly visual-based pseudo-haptic techniques, to increase availability as they provide feedback with reduced costs.

During the related work studies, we identified various concepts related to haptic feedback that we explored and disambiguated. Regarding Pseudo-Haptic techniques, we defined a classification to split the most common methods into similar groups. We then performed an analysis on what were the most relevant studies for each of them. Additionally, we investigated the material properties that each technique type was capable of providing feedback on and concluded which were their main strengths and weaknesses.

This analysis allowed us to realize that some techniques that showed great potential had not yet been further explored in Virtual Reality environments. With this in mind, we decided to join elements of the identified techniques, adapt them to a 3D environment, and focus on measuring the compliance property using exclusively visual feedback to achieve our goals.

In our approach, we proposed a deformation algorithm that is based on the Mass-Spring-Damper physical model combined with an adapted version of a Control/Display ratio technique that uses tracking information (provided by a LeapMotion controller) alongside an Inverse Kinematics algorithm that allows us to increase physical accuracy and realism further.

Subsequently, we proceeded with user experiments to evaluate whether or not this approach allowed users to receive accurate and realistic feedback that enables the perception of compliance. The evaluation process was comprised of three main tasks, with 16 trials each, with pairs of deformable objects. These combinations were categorized based on empirically chosen compliance levels. We requested users to identify which of the objects was the most compliant. We hypothesized that this technique would be able to provide quality feedback for visually equal objects,

objects with different scales, and with distinct textures. We also asked the users to give us as much feedback as possible, specifically focusing on the physical accuracy and realism of this approach.

We were able to conclude that this approach achieved positive results, as users were easily able to identify the shown compliance levels and considered the technique to be accurate and realistic. We did identify, however, that it is likely that the object's scale has an influence on the results of a couple situations, as users affirmed that they were more inclined to perceive a smaller object as more compliant.

Comparing our work to a recently published study, we can conclude that by employing similar algorithms, we were able to enhance the perception of compliance. We shared similar conclusions and were able to both add to their results as well as take some information from theirs. This shows that this area of research and this approach does, in fact, attract interest, as it proves to have a very high adaptability while still achieving positive results.

In our solution, it is possible to identify some limitations, such as the interaction being limited to the fingertips, the algorithm's efficiency not allowing the rendering of very high vertex objects, not being able to handle objects with complex compliance systems and the inverse kinematics asset used not including detailed constraints. Nonetheless, with all of this being said, we consider that we presented an approach that is able to provide accurate feedback on the property of compliance while being realistic, immersive, and straightforward.

5.1 Future Work

Although the results we presented are positive, we were still able to identify several limitations, most of them regarding efficiency. With that in mind, we consider that this approach is still largely unexplored, thus having vast room for improvement, which requires a considerable amount of additional research and testing. While we do subscribe to most of the future work proposed by Bouzbib et al. [10], we will mainly focus on the potential work to apply to the approach proposed in our study.

In order to enhance the level of interaction of our approach, we consider that being able to use both hands rather than just one is worth exploring in the future. However, to achieve that, an optimization of the current workload is required, and UltraLeap's software for tracking (or any other hand-tracking software being used) needs to be explored further.

Additionally, we consider that it is possible to increase the level of detail and realism by both allowing the user's entire hand and fingers to apply forces to the deformation system rather than just the fingertip and by implementing the respective constraints on the inverse kinematics algorithm. This does, however, have a severe effect on the computational load. Once again, in order to achieve that, a focus on optimization is required.

Furthermore, in our experimentation, we used a total of four different levels of compliance, two different scales, and two different textures. While Bouzbib et al. [10] does evaluate a wider range of values, we feel like it could be interesting enough to expand our research to achieve a more accurate pinpoint of compliance perception range using pseudo-haptics for all of these variables.

Finally, the combination of our visual-based pseudo-haptic technique with passive haptic devices or even day-to-day use objects is something very intriguing as it could potentially provide a near-perfect perception of touch feedback and thus profoundly increase the level of realism of a Virtual Reality environment.

References

[1] Merwan Achibet, Benoît Le Gouis, Maud Marchal, Pierre-Alexandre Léziart, Ferran Argelaguet, Adrien Girard, Anatole Lécuyer, and Hiroyuki Kajimoto. FlexiFingers: Multi-finger interaction in VR combining passive haptics and pseudo-haptics. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 103–106, March 2017.

[2] M. Ercan Altinsoy. The Effect of Auditory Cues on the Audiotactile Roughness Perception: Modulation Frequency and Sound Pressure Level. In Antti Pirhonen and Stephen Brewster, editors, *Haptic and Audio Interaction Design*, Lecture Notes in Computer Science, pages 120–129, Berlin, Heidelberg, 2008. Springer.

[3] Ferran Argelaguet, David Antonio Gómez Jáuregui, Maud Marchal, and Anatole Lécuyer. A Novel Approach for Pseudo-haptic Textures Based on Curvature Information. In Poika Isokoski and Jukka Springare, editors, *Haptics: Perception, Devices, Mobility, and Communication*, Lecture Notes in Computer Science, pages 1–12, Berlin, Heidelberg, 2012. Springer.

[4] Ferran Argelaguet, David Antonio Gómez Jáuregui, Maud Marchal, and Anatole Lécuyer. Elastic images: Perceiving local elasticity of images through a novel pseudo-haptic deformation effect. *ACM Transactions on Applied Perception*, 10(3):17:1–17:14, August 2013.

[5] Andreas Aristidou and Joan Lasenby. Fabrik: A fast, iterative solver for the inverse kinematics problem. *Graphical Models*, 73(5):243–260, 2011.

[6] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1968–1979, New York, NY, USA, May 2016. Association for Computing Machinery.

[7] Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. Controlling perceived stiffness of pinched objects using visual feedback of hand deformation. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 557–562, February 2014. ISSN: 2324-7355.

[8] Klaus-Jürgen Bathe. *Finite Element Method*, pages 1–12. John Wiley & Sons, Ltd, 2008.

[9] S. James Biggs and Mandayam A. Srinivasan. Haptic Interfaces. In *Handbook of Virtual Environments*. CRC Press, 2002. Num Pages: 24.

[10] Elodie Bouzbib, Claudio Pacchierotti, and Anatole Lécuyer. When Tangibles Become Deformable: Studying Pseudo-Stiffness Perceptual Thresholds in a VR Grasping Task. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2743–2752, May 2023. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[11] F.J. Canadas-Quesada and A. Reyes-Lecuona. Improvement of perceived stiffness using auditory stimuli in haptic virtual reality. In *MELECON 2006 - 2006 IEEE Mediterranean Electrotechnical Conference*, pages 462–465, May 2006. ISSN: 2158-8481.

[12] S. Chiaverini, O. Egeland, and R.K. Kanestrom. Achieving user-defined accuracy with damped least-squares inverse kinematics. In *Fifth International Conference on Advanced Robotics 'Robots in Unstructured Environments*, pages 672–677 vol.1, June 1991.

[13] Antoine Costes, Ferran Argelaguet, Fabien Danieau, Philippe Guillotel, and Anatole Lécuyer. Touchy : A Visual Approach for Simulating Haptic Effects on Touchscreens. *Frontiers in ICT*, 6, 2019.

[14] Jorge M. Cruz-Duarte, J. Juan Rosales-García, and C. Rodrigo Correa-Cely. Entropy generation in a mass-spring-damper system using a conformable model. *Symmetry*, 12(3), 2020.

[15] H. Das, J.-E. Slotine, and T.B. Sheridan. Inverse kinematic algorithms for redundant systems. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pages 43–48 vol.1, April 1988.

[16] L. Dominjon, A. Lecuyer, J.-M. Burkhardt, P. Richard, and S. Richir. Influence of control/display ratio on the perception of mass of manipulated objects in virtual environments. In *IEEE Proceedings. VR 2005. Virtual Reality, 2005.*, pages 19–25, March 2005. ISSN: 2375-5334.

[17] P. Edwards. Mass-spring-damper systems the theory, 2001.

[18] Giorgos Ganias, Christos Lougiakis, Akrivi Katifori, Maria Roussou, Yannis Ioannidis, and loannis Panagiotis Ioannidis. Comparing Different Grasping Visualizations for Object Manipulation in VR using Controllers. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2369–2378, May 2023. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[19] Steve Guest, Caroline Catmur, Donna Lloyd, and Charles Spence. Audiotactile interactions in roughness perception. *Experimental Brain Research*, 146(2):161–171, September 2002.

[20] Dustin T. Han, Mohamed Suhail, and Eric D. Ragan. Evaluating Remapped Physical Reach for Hand Interactions with Passive Haptics in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics*, 24(4):1467–1476, April 2018. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[21] Gunter Hannig and Barbara Deml. Scrutinizing pseudo haptic feedback of surface roughness in virtual environments. In *2008 IEEE Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 1–4, July 2008. ISSN: 1944-9410.

[22] Gunter Hannig, Barbara Deml, and Andreas Mihalyi. SIMULATING SURFACE ROUGHNESS IN VIRTUAL ENVIRONMENTS BY VIBRO-TACTILE FEEDBACK. *IFAC Proceedings Volumes*, 40(16):224–229, January 2007.

[23] Hsin-Ni Ho, Daisuke Iwai, Yuki Yoshikawa, Junji Watanabe, and Shin'ya Nishida. Combining colour and temperature: A blue object is more likely to be judged as warm than a red object. *Scientific Reports*, 4(1):5527, July 2014. Number: 1 Publisher: Nature Publishing Group.

[24] Brent Edward Insko. *Passive haptics significantly enhances virtual environments*. The University of North Carolina at Chapel Hill, 2001.

[25] HyeongYeop Kang, Geonsun Lee, and JungHyun Han. Visual Manipulation for Underwater Drag Force Perception in Immersive Virtual Environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 38–46, March 2019. ISSN: 2642-5254.

[26] Ben Kenwright. Inverse kinematics – cyclic coordinate descent (ccd). *Journal of Graphics Tools*, 16(4):177–217, 2012.

[27] Seung-Chan Kim, Ki-Uk Kyung, and Dong-Soo Kwon. The Effect of Sound on Haptic Perception. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, pages 354–360, March 2007.

[28] Luv Kohli. Redirected touching: Warping space to remap passive haptics. In *2010 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 129–130, March 2010.

[29] Luv Kohli, Mary C. Whitton, and Frederick P. Brooks. Redirected touching: The effect of warping space on task performance. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 105–112, March 2012.

[30] Ashok Kumar, Ravali Gourishetti, and M Manivannan. Mechanics of pseudo-haptics with computer mouse. In *2017 IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*, pages 1–6, October 2017.

[31] A. Lecuyer, S. Coquillart, A. Kheddar, P. Richard, and P. Coiffet. Pseudo-haptic feedback: can isometric input devices simulate force feedback? In *Proceedings IEEE Virtual Reality 2000 (Cat. No.00CB37048)*, pages 83–90, March 2000. ISSN: 1087-8270.

[32] Anatole Lécuyer. Simulating Haptic Feedback Using Vision: A Survey of Research and Applications of Pseudo-Haptic Feedback. *Presence*, 18:39–53, February 2009.

[33] Anatole Lécuyer, Jean-Marie Burkhardt, and Laurent Etienne. Feeling bumps and holes without a haptic interface: the perception of pseudo-haptic textures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 239–246, New York, NY, USA, April 2004. Association for Computing Machinery.

[34] Anatole Lécuyer, Jean-Marie Burkhardt, and Chee-Hian Tan. A study of the modification of the speed and size of the cursor for simulating pseudo-haptic bumps and holes. *ACM Transactions on Applied Perception*, 5(3):14:1–14:21, September 2008.

[35] K.E. MacLean. Designing with haptic feedback. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 1, pages 783–788 vol.1, April 2000. ISSN: 1050-4729.

[36] Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109–118, 2007.

[37] Takuji Narumi, Yusuke Ujitoko, Yuki Ban, Tomohiro Tanikawa, Koichi Hirota, and Michitaka Hirose. Resistive swipe: Visuo-haptic interaction during swipe gestures to scroll background images on touch interfaces. In *2017 IEEE World Haptics Conference (WHC)*, pages 334–339, June 2017.

[38] Akira Nomoto, Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. Supporting Precise Manual-handling Task using Visuo-haptic Interaction. In *Proceedings of the 7th Augmented Human International Conference 2016*, AH '16, pages 1–8, New York, NY, USA, February 2016. Association for Computing Machinery.

[39] Karljohan Lundin Palmerius, Daniel Johansson, Gunnar Höst, and Konrad Schönborn. An Analysis of the Influence of a Pseudo-haptic Cue on the Haptic Perception of Weight. In Malika Auvray and Christian Duriez, editors, *Haptics: Neuroscience, Devices, Modeling, and Applications*, Lecture Notes in Computer Science, pages 117–125, Berlin, Heidelberg, 2014. Springer.

[40] Maisarah Binti Ridzuan, Yasutoshi Makino, and Kenjiro Takemura. Direct Touch Haptic Display Using Immersive Illusion with Interactive Virtual Finger. In Poika Isokoski and Jukka Springare, editors, *Haptics: Perception, Devices, Mobility, and Communication*, Lecture Notes in Computer Science, pages 432–444, Berlin, Heidelberg, 2012. Springer.

[41] Majed Samad, Elia Gatti, Anne Hermes, Hrvoje Benko, and Cesare Parise. Pseudo-Haptic Weight: Changing the Perceived Weight of Virtual Objects By Manipulating Control-Display Ratio. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–13, Glasgow Scotland Uk, May 2019. ACM.

[42] Christian Sandor, Shinji Uchiyama, and Hiroyuki Yamamoto. Visuo-Haptic Systems: Half-Mirrors Considered Harmful. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, pages 292–297, March 2007.

[43] Mari Tatezono, Katsunari Sato, Kouta Minamizawa, Hideaki Nii, Naoki Kawakami, and Susumu Tachi. Effect of haptic feedback on pseudo-haptic feedback for arm display. In *2009 ICCAS-SICE*, pages 4332–4337, August 2009.

[44] Qianqian Tong, Wenxuan Wei, Yuru Zhang, Jing Xiao, and Dangxiao Wang. Survey on Hand-Based Haptic Interaction for Virtual Reality. *IEEE Transactions on Haptics*, 16(2):154–170, April 2023. Conference Name: IEEE Transactions on Haptics.

[45] Yusuke Ujitoko and Yuki Ban. Survey of Pseudo-Haptics: Haptic Feedback Design and Application Proposals. *IEEE Transactions on Haptics*, 14(4):699–711, October 2021. Conference Name: IEEE Transactions on Haptics.

[46] Yusuke Ujitoko, Yuki Ban, and Koichi Hirota. Modulating Fine Roughness Perception of Vibrotactile Textured Surface using Pseudo-haptic Effect. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):1981–1990, May 2019. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[47] Yusuke Ujitoko, Yuki Ban, and Koichi Hirota. Presenting Static Friction Sensation at Stick-slip Transition using Pseudo-haptic Effect. In *2019 IEEE World Haptics Conference (WHC)*, pages 181–186, July 2019.

[48] Yusuke Ujitoko, Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, Koichi Hirota, and Michitaka Hirose. Yubi-Toko: finger walking in snowy scene using pseudo-haptic technique on touchpad. In *SIGGRAPH Asia 2015 Emerging Technologies*, SA '15, pages 1–3, New York, NY, USA, November 2015. Association for Computing Machinery.

[49] Yannick Weiss, Steeven Villa, Albrecht Schmidt, Sven Mayer, and Florian Müller. Using Pseudo-Stiffness to Enrich the Haptic Experience in Virtual Reality. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–15, Hamburg Germany, April 2023. ACM.

Appendix A

Appendix

A.1 Participant Distributions

A.1.1 Gender

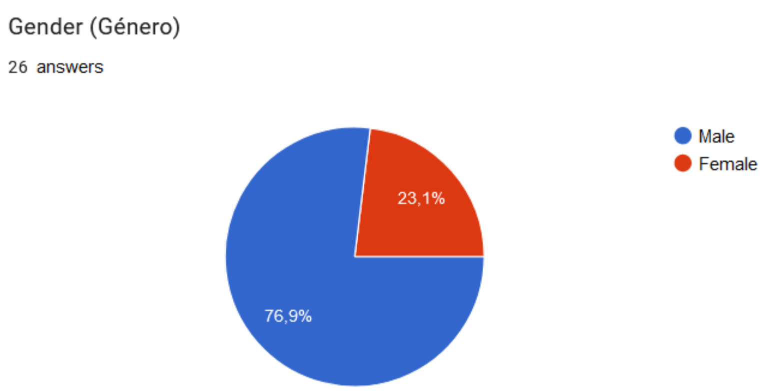


Figure A.1: Participant gender distribution.

A.1.2 Age

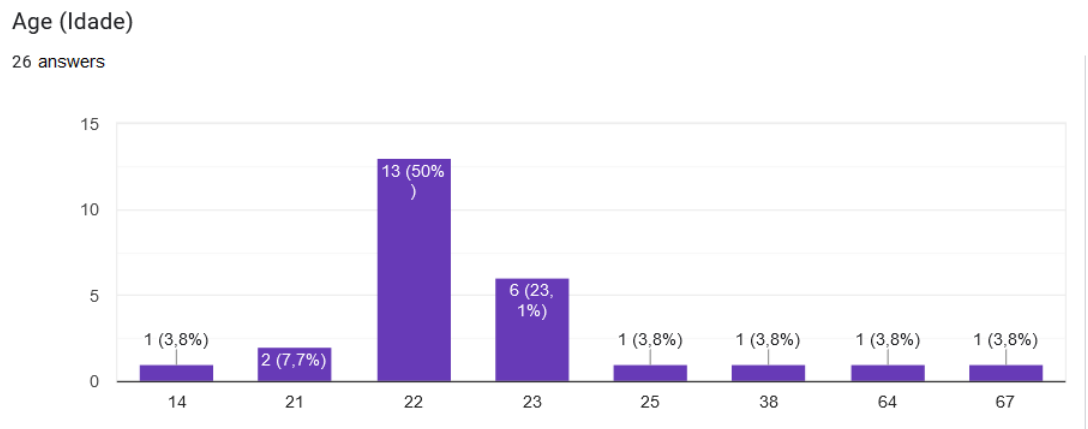


Figure A.2: Participant age distribution.

A.1.3 VR Experience

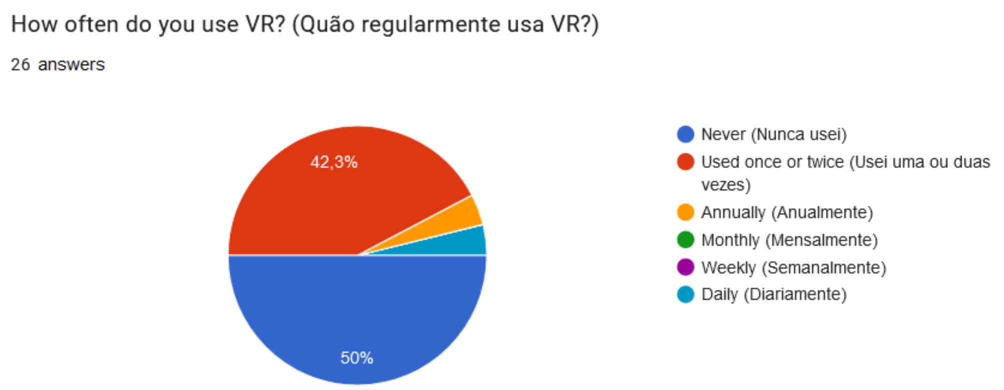


Figure A.3: Participant VR experience distribution.

A.1.4 Professions

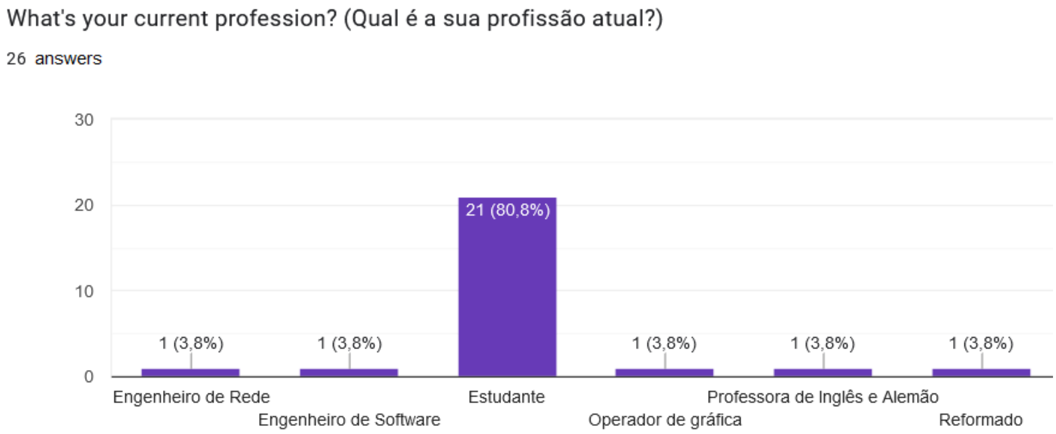


Figure A.4: Participant profession distribution.

A.1.5 Habilitations

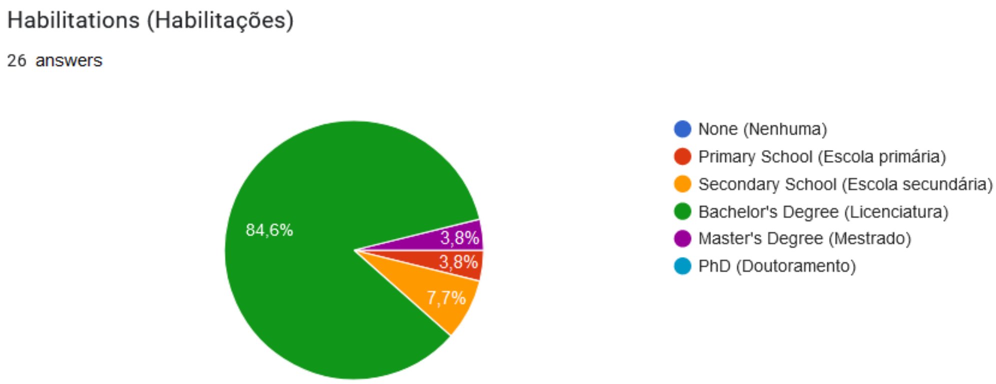
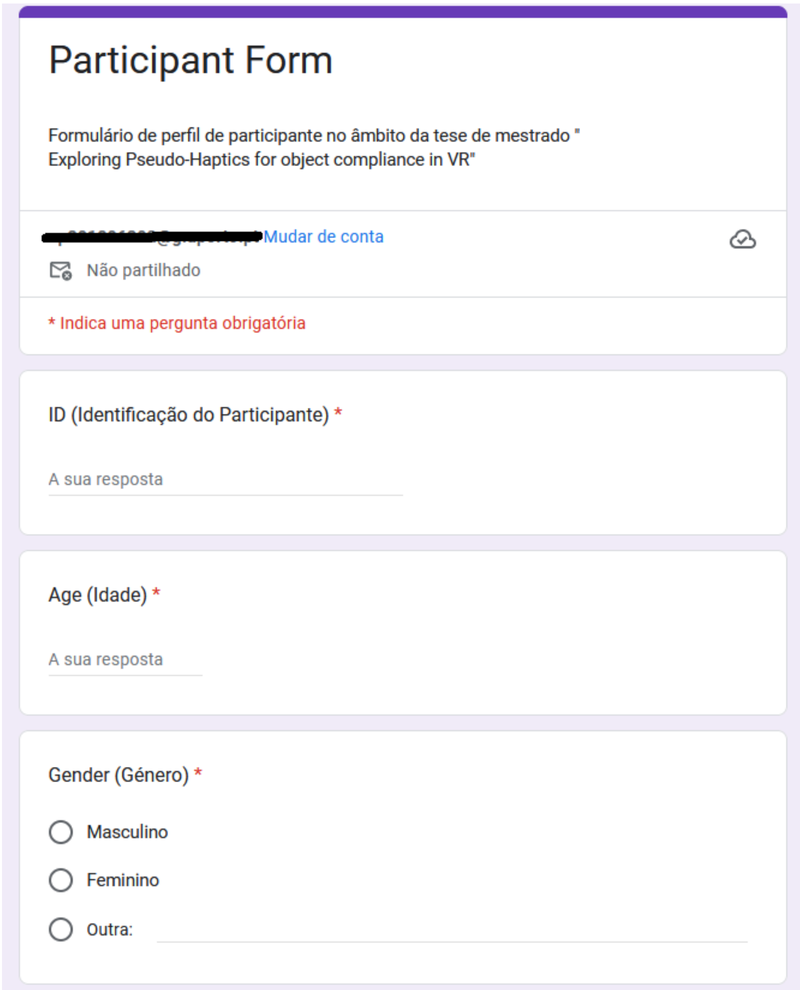


Figure A.5: Participant habilitation distribution.

A.2 Forms

A.2.1 Participant Form


Participant Form

Formulário de perfil de participante no âmbito da tese de mestrado " Exploring Pseudo-Haptics for object compliance in VR"

[redacted] Mudar de conta

Não partilhado

* Indica uma pergunta obrigatória

ID (Identificação do Participante) *

A sua resposta

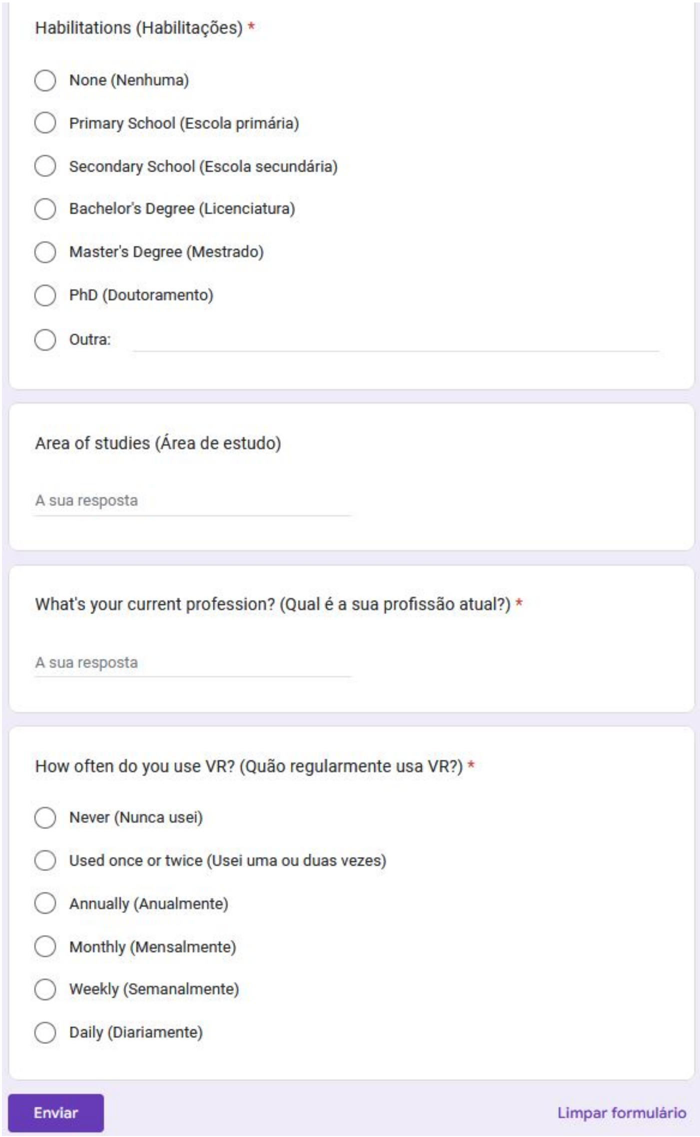
Age (Idade) *

A sua resposta

Gender (Género) *

- Masculino
- Feminino
- Outra:

Figure A.6: Participant form part 1.



Habilitations (Habilitações) *

- None (Nenhuma)
- Primary School (Escola primária)
- Secondary School (Escola secundária)
- Bachelor's Degree (Licenciatura)
- Master's Degree (Mestrado)
- PhD (Doutoramento)
- Outra:

Area of studies (Área de estudo)

A sua resposta

What's your current profession? (Qual é a sua profissão atual?) *

A sua resposta

How often do you use VR? (Quão regularmente usa VR?) *

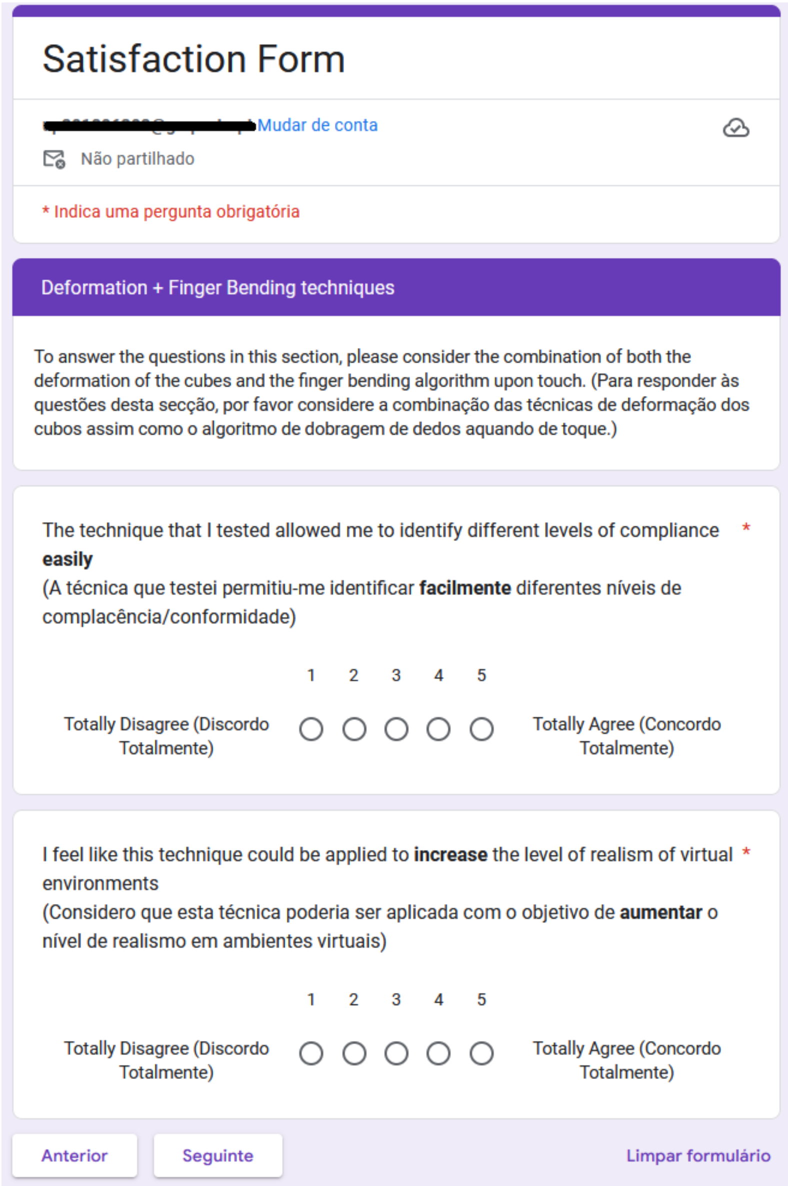
- Never (Nunca usei)
- Used once or twice (Usei uma ou duas vezes)
- Annually (Anualmente)
- Monthly (Mensalmente)
- Weekly (Semanalmente)
- Daily (Diariamente)

Enviar

Limpar formulário

Figure A.7: Participant form part 2.

A.2.2 Satisfaction Form



Satisfaction Form

[redacted] Mudar de conta

Não partilhado

* Indica uma pergunta obrigatória

Deformation + Finger Bending techniques

To answer the questions in this section, please consider the combination of both the deformation of the cubes and the finger bending algorithm upon touch. (Para responder às questões desta secção, por favor considere a combinação das técnicas de deformação dos cubos assim como o algoritmo de dobragem de dedos aquando de toque.)

The technique that I tested allowed me to identify different levels of compliance **easily** *
(A técnica que testei permitiu-me identificar **facilmente** diferentes níveis de complacência/conformidade)

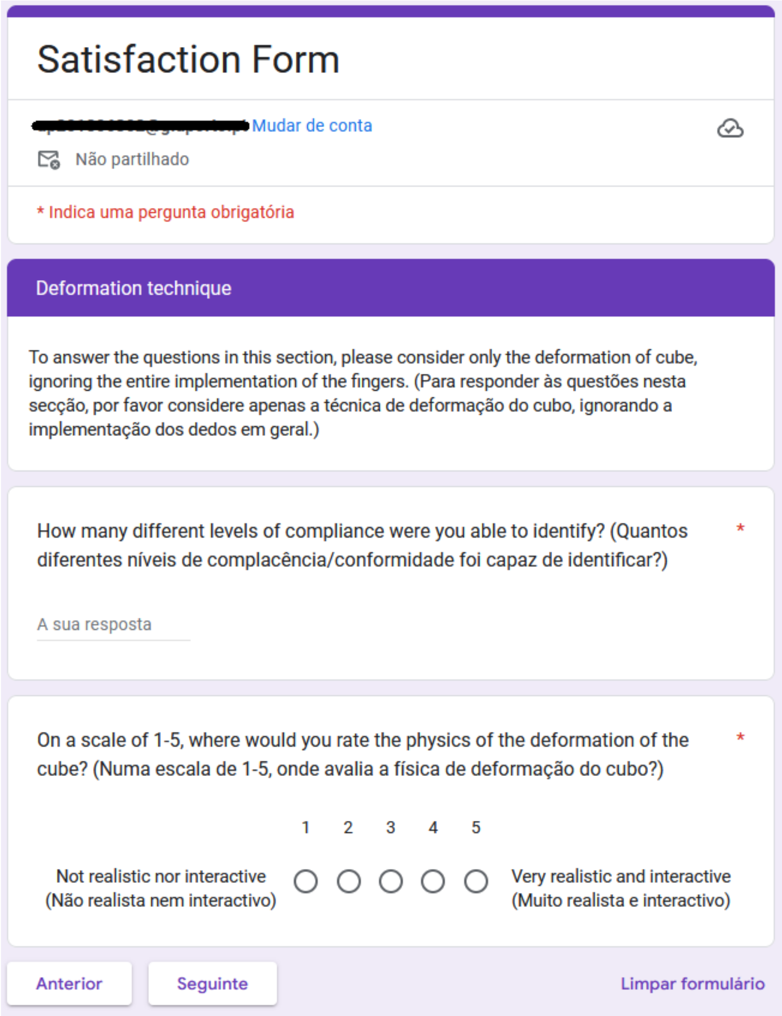
	1	2	3	4	5	
Totally Disagree (Discordo Totalmente)	○	○	○	○	○	Totally Agree (Concordo Totalmente)

I feel like this technique could be applied to **increase** the level of realism of virtual environments *
(Considero que esta técnica poderia ser aplicada com o objetivo de **aumentar** o nível de realismo em ambientes virtuais)

	1	2	3	4	5	
Totally Disagree (Discordo Totalmente)	○	○	○	○	○	Totally Agree (Concordo Totalmente)

Anterior Seguinte Limpar formulário

Figure A.8: Satisfaction form part 1.



Satisfaction Form

[redacted] Mudar de conta

Não partilhado

* Indica uma pergunta obrigatória

Deformation technique

To answer the questions in this section, please consider only the deformation of cube, ignoring the entire implementation of the fingers. (Para responder às questões nesta secção, por favor considere apenas a técnica de deformação do cubo, ignorando a implementação dos dedos em geral.)

How many different levels of compliance were you able to identify? (Quantos diferentes níveis de complacência/conformidade foi capaz de identificar?) *

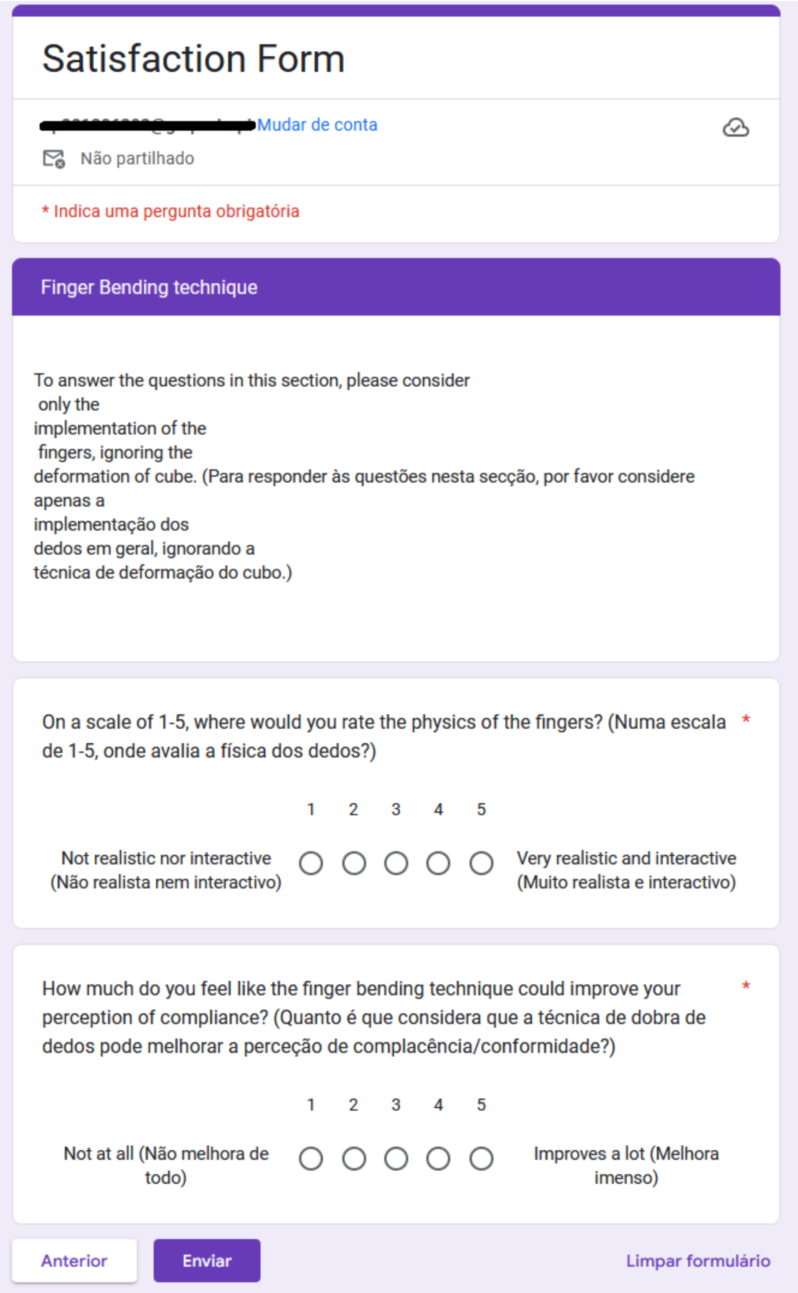
A sua resposta

On a scale of 1-5, where would you rate the physics of the deformation of the cube? (Numa escala de 1-5, onde avalia a física de deformação do cubo?) *

	1	2	3	4	5	
Not realistic nor interactive (Não realista nem interactivo)	○	○	○	○	○	Very realistic and interactive (Muito realista e interactivo)

Anterior Seguinte Limpar formulário

Figure A.9: Satisfaction form part 2.



Satisfaction Form

[redacted] Mudar de conta

Não partilhado

* Indica uma pergunta obrigatória

Finger Bending technique

To answer the questions in this section, please consider
only the
implementation of the
fingers, ignoring the
deformation of cube. (Para responder às questões nesta secção, por favor considere
apenas a
implementação dos
dedos em geral, ignorando a
técnica de deformação do cubo.)

On a scale of 1-5, where would you rate the physics of the fingers? (Numa escala de 1-5, onde avalia a física dos dedos?) *

	1	2	3	4	5	
Not realistic nor interactive (Não realista nem interactivo)	○	○	○	○	○	Very realistic and interactive (Muito realista e interactivo)

How much do you feel like the finger bending technique could improve your perception of compliance? (Quanto é que considera que a técnica de dobra de dedos pode melhorar a perceção de complacência/conformidade?) *

	1	2	3	4	5	
Not at all (Não melhora de todo)	○	○	○	○	○	Improves a lot (Melhora imenso)

Anterior Enviar Limpar formulário

Figure A.10: Satisfaction form part 3.

A.3 Satisfaction Answers

A.3.1 Question 1.1.

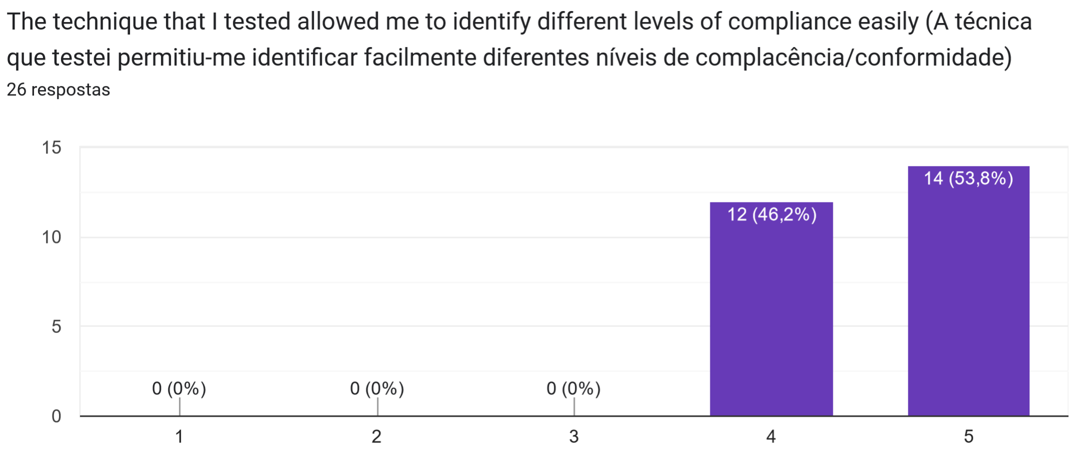


Figure A.11: Bar chart of the answers to question 1.1.

A.3.2 Question 1.2.

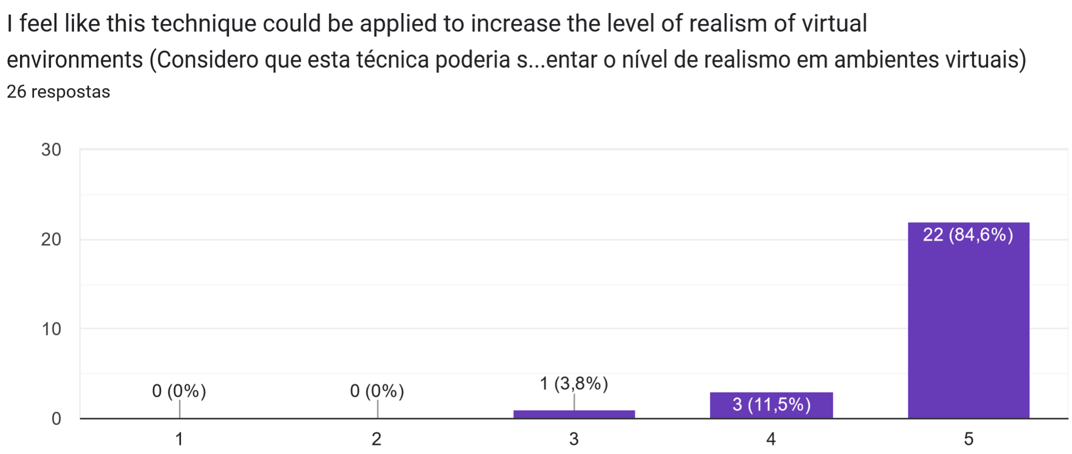


Figure A.12: Bar chart of the answers to question 1.2.

A.3.3 Question 2.1.

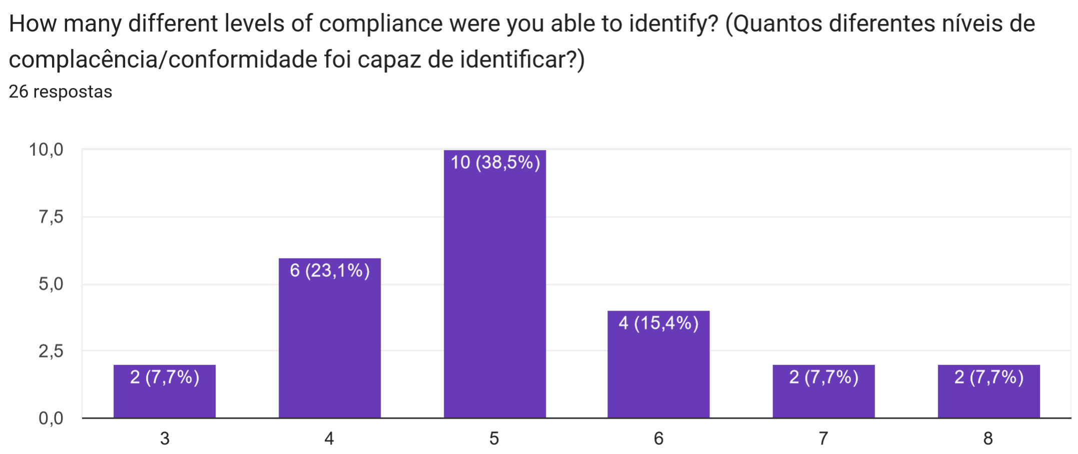


Figure A.13: Bar chart of the answers to question 2.1.

A.3.4 Question 2.2.

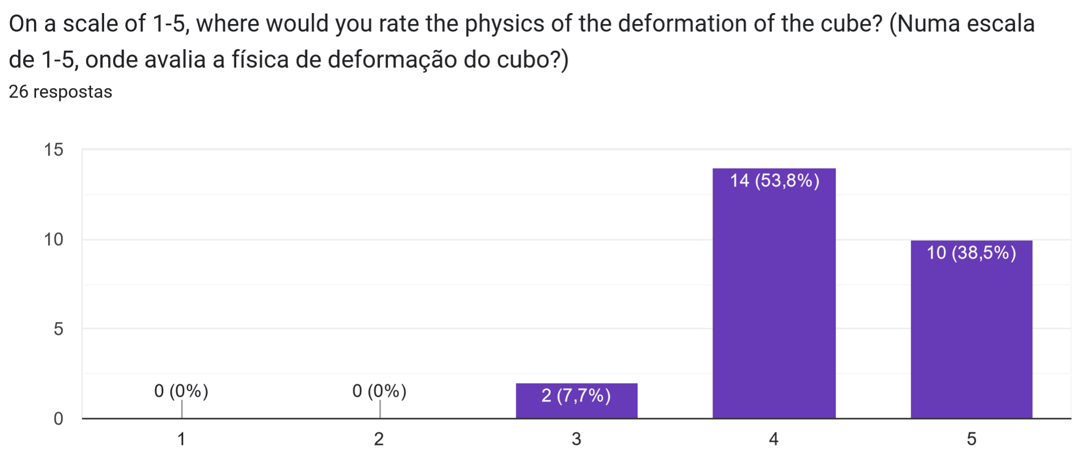


Figure A.14: Bar chart of the answers to question 2.2.

A.3.5 Question 3.1.

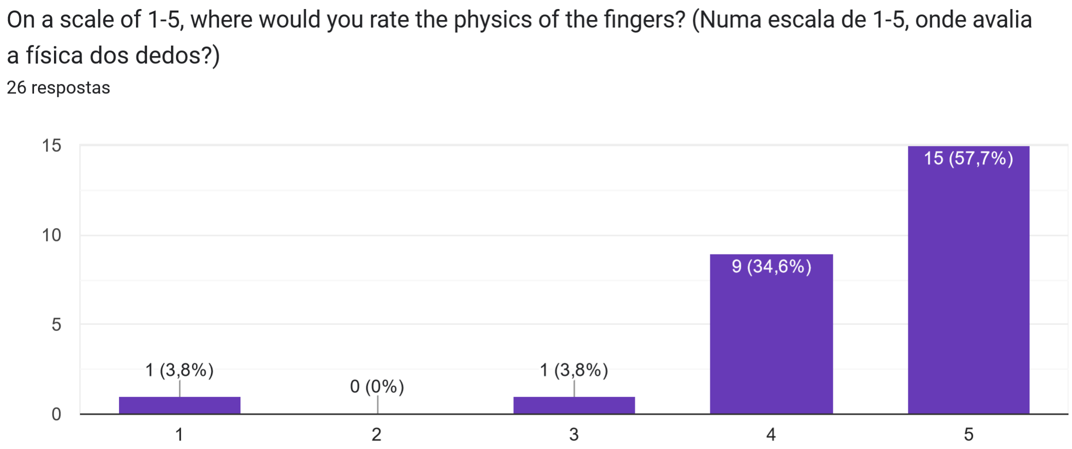


Figure A.15: Bar chart of the answers to question 3.1.

A.3.6 Question 3.2.

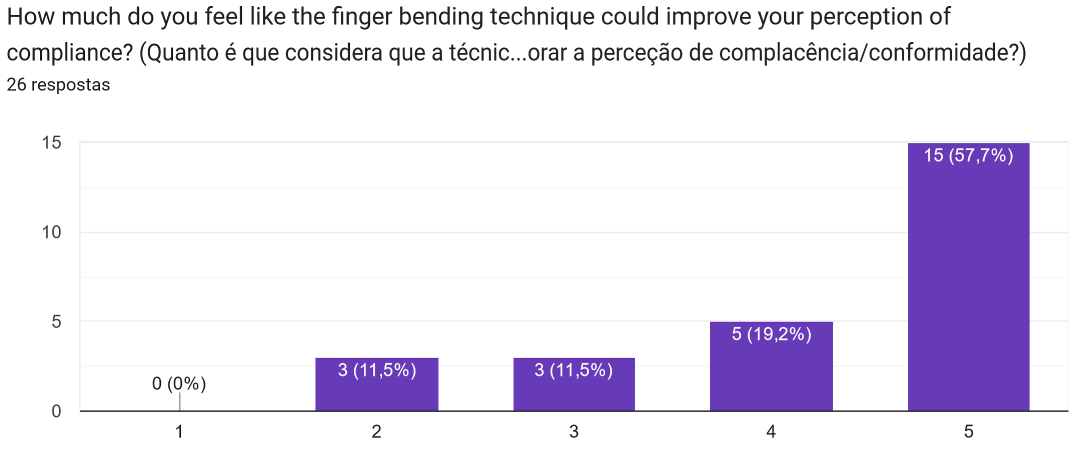


Figure A.16: Bar chart of the answers to question 3.2.