Bachelor of Science in Digital Game Development
June 2020

A Study on the Perceived Realism Of Strand-based Hair Simulated By Style

Evaluating Real-time Hair-simulation in Unreal Engine 4

Jennifer Larsson-Ståhl

Faculty of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfilment of the requirements for the degree of Bachelor of Science in Digital Game Development. The thesis is equivalent to 10 weeks of full time studies.

The authors declare that they are the sole authors of this thesis and that they have not used any sources other than those listed in the bibliography and identified as references. They further declare that they have not submitted this thesis at any other institution to obtain a degree.

Contact Information:
Author(s):
Jennifer Larsson-Ståhl
E-mail: jesh17@student.bth.se

University advisor:
Assoc. Prof. Veronica Sundstedt
Department of Computer Science

Faculty of Computing
Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

Abstract

Background. In order to increase the visual fidelity of characters in video games, strand-based hair rendering within Unreal Engine 4 is explored in this thesis. Because of the complex nature of hair, explicit hair models can be very costly to render and simulate. It is theorized that utilizing customized hair and simulation settings, tailored to specific types of hairstyles could alleviate this issue while preserving visual fidelity.
Objectives. The aim of this thesis is to provide an insight into what can be done to increase visual appeal and computer performance of physical simulation for different types of hairstyles and determine if customized setting may cause a significant impact on the perceived level of realism. The objectives of this thesis are to acquire a set of different strand-based hairstyles, determine a set of customized hair and simulation settings which can be applied to them, create a test scene inside of Unreal Engine 4 and render out a set of images and videos to be used in a user experiment, measure the performance of each customized setting and finally synthesize the acquired data.
Methods. In order to achieve the aims and objectives of this thesis, a user experiment that utilizes the 2AFC method to let participants compare image- and video-pairs is performed as well as a performance experiment using the built-in profiling tools in Unreal Engine 4. In addition, a pilot experiment was performed in order to ascertain that the experiments would be feasible on the available hardware.
Results. The results show that there was a significant difference in the perceived level of realism when different simulation settings was applied to the hairstyles, with customized settings being preferred to the default setting. The voting results on the image-pairs showed a preference for fine hair strands while the strand count did not have as much of an impact. It was shown that participants could easier distinguish between the different simulation- and hair-settings in long hair compared to short hair. The performance experiment showed that the amount of hair strands had the biggest impact on computer performance.
Conclusions. Customizing hair and simulation settings to different types of hairstyles could provide a heightened perceived level of realism and a limited performance boost, from what could be derived from these experiments. Lowering the hair strand count was determined to be the most effective method of increasing performance. The performance of strand-based hair is currently not quite reaching a consistent 60 frames per second on the tested hardware, but with further optimizations it is believed that this could be acquired, especially on more powerful graphics cards. Future work should keep focus on increasing the stability of real-time strand-based hair simulation.

Keywords: computer graphics, photo-realism, physical simulation, real-time hair rendering, unreal engine 4

Acknowledgments

The author of this thesis would like to thank their university supervisor Veronica Sundstedt for their supervision and guidance throughout the work on this thesis. The author would also like to thank Francisco Lopez Luro for sharing valuable comments on the execution of the performance profiling. Lastly the author would like to thank all of the participants who participated in the user experiment.

Contents

Abstract **i**

Acknowledgments **iii**

1 Introduction **1**
1.1 Aims and objectives 2
1.2 Research questions and hypothesis 3
1.3 Limitations 3

2 Related Work **5**
2.1 Hair simulation and rendering 5
2.2 Visual perception of realism 6

3 Background theory **7**
3.1 Hair simulation overview 7
3.2 Unreal Engine 4 8
3.2.1 Hair rendering and simulation within Unreal Engine 4 8
3.2.2 Definitions 9

4 Method **11**
4.1 Pilot experiment 11
4.2 Environment setup 12
4.2.1 Asset creation 12
4.2.2 Simulation and hair settings 14
4.2.3 Image and video rendering 16
4.3 User experiment 16
4.3.1 Image- and video-pairs 17
4.3.2 Ethical considerations 18
4.4 Performance profiling 18
4.4.1 Enabled console commands 19

5 Results and Analysis **21**
5.1 User experiment results 21
5.2 Performance results 24

6 Discussion **27**
6.1 Visual impact of customized settings 27
6.2 Performance impact of customized settings 29

6.3 Viability of results 29

7 Conclusions and Future Work **31**
7.1 Conclusions 31
7.2 Future work 32

References **35**

A Performance data for simulation settings **37**

B Performance data on different hairstyle strand count **39**

C Detailed vote distributions **41**

D Invitational letter **43**

List of Figures

4.1 The scene used during the experiments 12
4.2 Failure case: Braid not staying together during simulation 14
4.3 Import settings for grooms 15
4.4 All of the images included in the user experiment 16
4.5 Frame sequence from one of the video clips used in the user experiment 17
4.6 Illustration describing how hair rendering settings were compared against each other 18
4.7 Illustration describing how simulation settings were compared against each other 18

5.1 Vote distribution on image-pairs 21
5.2 Vote distribution on video-pairs 22

A.1 Frame time (ms) 37
A.2 GPU total time (ms) 37
A.3 Game thread time (ms) 38
A.4 Render thread time (ms) 38
A.5 Niagara GPU simulation time (ms) 38

B.1 Frame time (ms) 39
B.2 GPU total time (ms) 39
B.3 Game thread time (ms) 40
B.4 Render thread time (ms) 40
B.5 Niagara GPU simulation time (ms) 40

C.1 Chart showing vote distribution on all image-pairs 41
C.2 Chart showing vote distribution on all video-pairs 41

List of Tables

4.1 Created hairstyles 13
4.2 Hair settings 15
4.3 Simulation settings 16

5.1 Still-image evaluation results sorted by p-value 23
5.2 Video evaluation results sorted by p-value 24
5.3 Performance measurements for different hair simulation settings (ms) 25
5.4 Performance measurements for different hair strand counts (ms) . . . 26

Chapter 1

Introduction

Video games are becoming an increasingly popular medium for storytelling, and thus a lot of attention is going toward increasing the visual fidelity of game environments and characters in order to create immersive experiences for players to enjoy. Characters play a central role in the narration of a story but creating truly believable digital humans can be a challenging task. There are many aspects that contributes to the overall perceived level of realism in a character, one of them being the visual appearance of hair.

The complex nature of hair makes real-time simulation of photo-realistic hair a long standing challenge in computer graphics [22]. A human scalp generally contains about a hundred thousand strands of hair [22] which are all very thin and semi-translucent. Both of these properties makes it difficult to render hair using conventional methods, especially in real-time. Several different models exist for representing hair and these can be classified as explicit hair models, cluster hair models and volumetric textures [16]. Each excels within different areas. While explicit hair models provides a close to real-life representation of hair in the form of strands, cluster hair models aims to simplify the overall hair geometry by clustering several hair strands together to reduce computational costs while also making it more manageable. Implicit models that make use of volumetric textures mainly excels at representing short, fuzzy hair such as fur [14, 2].

Explicitly rendering and simulating individual hair strands, while providing an accurate representation of hair, commonly requires the execution of computationally heavy operations which would impact the performance of an application. This has lead to the development of several methods that aims to increase performance by simplifying the hair geometry while preserving the appearance of individual strands [22]. The most commonly used method of representing hair within video games is by the use of textured planes called hair-strips or hair cards. Each hair card represents a group of hair strands and the illusion of individual strands is created by rendering parts of the card as transparent with a technique called alpha-blending. This method makes it possible to render entire volumes of hair strands without significantly impacting performance, but simulation is limited to affecting clusters of hair in comparison to explicit hair models where every individual strand can be physically simulated. This makes it difficult to physically simulate card-based hairstyles in a natural-looking way. In addition to this, in order for hairstyles to appear convincing and non-two-dimensional, a significant amount of artistic skill is required during the authoring of card-based hairstyles. Due of these reasons, explicit (strand-based) representations of hair tend to be preferred in situations where limitations on render-time do not apply to a greater extent, such as in film production. In order to increase the visual fidelity of video game characters, real time strand-based hair simulation needs to be further explored.

It was not until quite recently that real-time strand-based hair rendering became more accessible to game developers. One example of this can be seen in *HairWorks*, a toolset developed by Nvidia which has been utilized in the production of the video game *The Witcher 3: Wild Hunt* (CD Projekt RED, 2015) [24] and another example would be *TressFX* developed by AMD which was utilized for the simulation and rendering of the main character's hair in the video game *Tomb Raider* (Square Enix, 2013) [17]. In the most recent updates of the openly available game engine *Unreal Engine 4* the ability to simulate strand-based hair in real-time was introduced as well [7]. While this addition to the engine is still in beta, it has made this kind of technology accessible to more developers as well as hobbyists than ever before.

Although the mentioned implementations have generally shown impressive visual results when applied to generic hairstyles, hair can be styled in many different ways such as buns, ponytails and braids and be of different types such as straight, wavy or curly. In addition, the current implementations of real-time strand-based hair are still in their early stages and prone to performance issues. The visual result and computer performance of simulated hair might vary greatly depending on hair thickness, hair length, style and simulation methods as shown by Jung and Lee [13]. This raises questions about whether creating customized simulation and rendering settings for certain kinds of hairstyles using the aforementioned implementations, could increase the perceived level of realism in strand-based hair while also alleviating performance issues. This is the topic which is explored in this thesis.

By conducting a user experiment that collects data on how participants perceive an assortment of different hairstyles with different simulation settings applied, this thesis hopes to determine important factors that contribute to the perceived level of realism in strand-based hair simulation. By combining the data collected from the user experiment with performance tests, the study aims to provide general guidelines for how to increase the accuracy of physical simulation for different types of hairstyles without sacrificing computer performance.

The chosen environment for this study is *Unreal Engine 4* (UE4). This game engine was chosen in order to explore its' experimental strand-based hair rendering features as well as being the most widely available option when it comes to the topic of this thesis.

1.1 Aims and objectives

The aim of this thesis is to explore the simulation and rendering of real-time strand-based hair within Unreal Engine 4 and provide an insight into what can be done to increase the visual appeal and computer performance of physical simulation for different types of hairstyles. By determining if different hair and simulation settings may have a significant impact on the perceived level of realism when applied to certain types of hairstyles, optimized settings can be acquired when combined with performance data. The performance data is collected by utilizing the built-in profiling tools within UE4 while a user experiment is conducted to collect data on visual perception.

The aim of this thesis hopes to be achieved by completion of the following objectives:

- Acquire a selection of different strand-based hairstyles that are supported by Unreal Engine 4.
- Determine appropriate hair and simulation settings that can be tested and applied to the different strand-based hairstyles.
- Import each hairstyle into a predefined scene inside Unreal Engine 4 and apply the acquired settings in order to render out images and video to be used in a user experiment.
- Conduct an online experiment where participants compares the level of realism of the different settings for the different hairstyles using the prepared images and videos.
- Measure the performance of the different settings applied to the different hairstyles using the profiling tools within Unreal Engine 4.
- Synthesize the acquired data.

1.2 Research questions and hypothesis

This thesis aims to answer the following research questions:

- What simulation settings inside Unreal Engine 4 produces the most realistic results for a set of different types of strand-based hairstyles?
- How can different hair simulation settings for strand-based hair be used to optimize computer performance without sacrificing fidelity?

It was hypothesized by the author that it is possible to achieve better computer performance and a heightened perceived level of realism by applying customized simulation settings to certain types of hairstyles. This is due to hair possessing many properties that may vary, such as curliness and length which results in different physical behaviour in real life. Additionally, people might have varying expectations on what certain types of hairstyles should look like when simulated, and this might not always be coherent with what developers expect, perhaps even resulting in an opportunity to choose a computationally cheaper alternative for simulation when a noticeable difference cannot be detected by the general public.

1.3 Limitations

There are certain limitations that had to be considered during this project. This project was executed during the covid-19 outbreak of 2020 which classified as a global pandemic at the time. These circumstances required people to socially distance themselves from each other which created some difficulty in conducting controlled user experiments. This had to be considered in this case. It was therefore decided that the user experiment would be performed via filling in an online form.

The educational institution which this bachelor thesis is being performed at was closed to all students throughout the entirety of the time period in which this thesis was performed. This has mainly limited the access to hardware that could be used for the performance tests. Performance tests could only be performed on one computer for this reason.

There are also time restrictions to consider. Due to this being a bachelor thesis project, a limited amount of time could be allocated to the creation of hair assets and the execution of experiments. All possible simulation and hair settings could not be explored in depth for this reason.

Lastly, the acquired results from the experiments are designed to only be applied to UE4. While some results might not exclusively apply to a certain game engine, application within other game engines lies outside the scope of this thesis.

Chapter 2

Related Work

A brief overview of works that are related to the topic of this thesis are presented in this chapter.

2.1 Hair simulation and rendering

A lot of research has been made regarding the simulation and rendering of photo-realistic real-time hair. The contributions in the area are far too numerous for all to be included here. Ward et al. [22] provides one of the most extensive overviews of the area as a whole to date by performing a literature review on the topic of hair styling, simulation and rendering. The study involves papers published up until year 2007 within the area. The survey provides a good introduction to the concept of hair rendering and simulation, as well as a thorough evaluation of the current methods for achieving realistic-looking digital hair. While some of the brought up methods can be considered more or less obsolete by now, many of the issues that are brought up regarding real-time performance still remain today. While the work provides a great overview, it does not conduct any experiments in order to solve any of the brought up problems, contrary to this thesis.

Before the start of this project, a systematic literature review on the topic of hair simulation and rendering was performed as well. This was done in order to find relevant information in preparation for this study. The study found that strand-based hair rendering had not been greatly utilized within video game production, mainly due to performance issues. It was concluded in the study that hair cards were a popular solution to the performance problems, but strand-based approaches could provide a more accurate representation of hair. This greatly inspired the decision of exploring strand-based hair rendering and simulation in this thesis.

The work performed by Jung and Lee [13] was a big source of inspiration for the topic of this thesis as well. In their work, methods of increasing the performance of real-time hair simulation by the application of hairstyle-specific simulation methods are presented. In their approach, an algorithm was developed that classifies different clusters of hair and applies simulation methods and parameter settings accordingly. The results were generally positive but a few failure cases were found where their method could not be applied. Failure cases resulted in clusters being wrongly classified by their algorithm. Their method is also limited by only being able to classify four different types of hairstyles. It is therefore recommended in their work to explore a wider variety of hairstyles in future research, one example being styles specific to natural African hair. This is attempted in this thesis and one such hairstyle is included in the experiments performed.

Other methods of increasing the performance of real-time hair rendering has been explored as well. One such is described in the work by Jansson et al. [12] which proposes a hybrid method as a form of level of detail optimization for strand-based hair. Level-of-detail optimizations generally involves some type of simplification that activates depending on some variable such as player distance from an object. It is noted that simply making the hair strands fewer and thicker at a greater distance to increase performance could accidentally remove too much detail and considering this, a hybrid rendering method is proposed instead. This means switching between a strand-based rasterizer which is used for close-up shots and a volume-based approximation of the rasterizer's output when the camera distance is greater. Two approaches are compared in their work and the performance is measured for both. Their results showed that their approach could provide a performance increase as well as the ability to be easily implemented into already existing strand-based pipelines. It is mentioned however, that their approach does not work well for all cases, which still makes a study in the topic of hairstyle-specific optimizations quite relevant.

2.2 Visual perception of realism

There has been some earlier work done regarding the visual perception of realism in digital characters and hair, but the topic of perceived realism in digital hair simulation has not been greatly explored. There is one study by Ramesh et al. [21] which evaluates hair shine by performing three separate experiments which evaluates the perceived level of realism, health, naturalness, and shine in physically rendered hair. The study is performed using virtual hair instead of real hair in order to more easily be able to control different properties of the hair through parameterization. The study does not take hair simulation or animation into consideration, but uses a similar approach for the experiments as proposed for the user experiment in this thesis. Their results showed that certain variables such as hair color and strand orientation impacted the perceived realism of hair more than others. The achieved results were backed up by earlier studies on real hair and they could therefore determine digital gloss experiments on virtual hair as being reliable enough to be a viable replacement for using real hair in similar experiments. Another thing which is mentioned by Ramesh et al. [21] is that visual perception can be affected by many different factors. In the work by McDonnell et al. [18] one such factor, namely the shading, is being investigated in relation to the perceived realism of human faces. In their work, a series of psycho-physical experiments are performed to determine the perceived realism of a real-time rendered character with different shading styles applied to it. This is done in order to provide developers with guidelines for how to visually present digital characters in a pleasant manner. Both still-images and video was used for their experiments and participants were asked to rank the realism, appeal, familiarity, re-assurance, friendliness and trustworthiness of the stimuli on a scale. Motion capture data for body, face and eye movement was applied to the character in order for the animations to seem as realistic as possible. The experiments found highly realistic styles as well as cartoon-styles to be given the highest positive ratings, but styles which fell somewhere in-between were given negative ratings.

Chapter 3

Background theory

The following chapter provides relevant background theory that is helpful for the understanding of the methods, results and discussion in this thesis. An overview of strand-based hair simulation is provided before proceeding to explain some of the specifics of hair simulation within UE4. Some information about the game engine in general are also given and lastly, a list of definitions is presented.

3.1 Hair simulation overview

Strand-based hair simulation has not been greatly utilized in video game development. The reasons for this being that hair is naturally very complex which results in performance difficulties when attempting to simulate in real-time. Some of the common approaches to alleviating these difficulties are presented in this section.

Common problems which makes strand-based hair computationally expensive to render regard anti-aliasing methods, transparency algorithms and the multiple scattering properties of hair strands [23]. Since hair strands are very thin, usually sub-pixel size, they are prone to aliasing artifacts which can be remedied by applying one of the numerous existing anti-aliasing algorithms. Transparency issues can be alleviated by implementing order-independent-transparency techniques or by depth-sorting strands, which is generally considered an expensive operation. Lastly, there are several shading models which can be used to approximate multiple scattering in hair strands, all which are generally computationally heavy. By using a shading model which omits the multiple scattering component, it is possible to significantly speed up computation.

A common approach to rendering hair strands is to render them as line segments which are expanded into camera-facing billboards. This allows the strand thickness to be dynamically changed, or the application of a two-dimensional texture map. The line segments are made from several vertices that creates a chain which can be physically simulated by several methods. The vertices can also be referred to as control points.

Approaches to hair simulation shares many similarities with cloth simulation in the sense that they all involve small geometric features being simulated, and require robust collision-treatment [4]. In order to simulate hair strands in real-time, guide strands are commonly used instead of simulating each strand by itself. Additional hair strands are then created on the GPU by interpolating between the guide strands. This is done in order to speed up rendering and simulation [23]. Interpolating the hair strands can however create collision problems, since the interpolated strands are simply copied geometry, appearing in a new location. This may result in hair strands intersecting with other geometry, which creates a very unnatural look. This issue can be alleviated by the use of adaptive skinning methods for hair, which interactively chooses which guide strands that interpolated strands should follow [4]. Alternatively, the dynamic follow-the-leader (DFTL) approach could be used in combination with velocity correction [19] as well to simulate one of the vertices and have the rest of the vertices of the hair strand simply follow.

When it comes to applying the actual physics to strand-based hairstyles, position-based methods can be used favorably. These methods are significantly cheaper to compute than classical physics simulation methods and particularly tailored for use in interactive environments [3]. An extended version of position-based-dynamics (XPBD) which solves some of the most well-known limitations of the method has been presented by Macklin et al. [15] which further increases this method's usability. When it comes to realistic hair simulation, it is important to consider hair-hair interaction and shape-preservation as well. This can be implemented into XPBD as constraints.

3.2 Unreal Engine 4

Unreal Engine 4 is a game engine developed by Epic Games. The engine is well-known within the video game industry for providing high-end visuals as well as a straightforward user interface. Wile the engine has been used for the development of several big commercial game projects, the engine is free to use for non-commercial purposes as well. There is extensive online documentation available for the engine as well as many of the built-in functionality receiving regular updates. These are some of the reasons why UE4 was chosen for this project.

While source code is available to developers, the engine is not open-source. For this reason, there are some details regarding the inner workings of the engine that are not known and therefore cannot be shared in this thesis.

3.2.1 Hair rendering and simulation within Unreal Engine 4

All details regarding the strand-based hair simulation and rendering in UE4 are not fully explained within the engine documentation, but their approach utilizes many of the already-known methods described in section 3.1. There are some specifics that will be explained in this section as well. During the writing of this thesis, strand-based hair rendering and simulation within UE4 is currently still considered a beta-feature of the engine. This means that Epic Games does not recommend developers to ship games with these features enabled. The feature has however been made available for testing purposes.

The engine differentiates the strand-based hair asset from other assets by referring to it as a "groom actor" or simply "groom". The simulation of grooms is based on their built-in Niagara VFX-system [7], which is a programmable VFX-system used primarily for simulating particle effects. The simulation is driven by one of the available physics solvers, created with Niagara. Physics solvers can also be customized or created from scratch, but for the sake of simplicity, this feature is not explored in this thesis. The groom asset also contains several parameters which can be customized, both for the rendering and simulation of hair. A complete reference to these parameters can be found in the UE4 documentation [9]. This thesis has chosen to focus on the material constraint parameters and the strand parameters in particular. The material constraints include bending, stretching and collision while the strand parameters decide particle attributes for simulation.

3.2.2 Definitions

This sections contains a short list of explanations of words and abbreviations associated to Unreal Engine 4. These have been included to provide easier understanding throughout the rest of this thesis.

- **Alembic**: A general file format that can be used to store arbitrary data as well as geometry and animation data.
- **Asset**: Content that can be used by a game engine is commonly referred to as an asset.
- **DCC tool**: Digital content creation tool. In this case it refers to a piece of software used for the authoring of video game assets.
- **Groom**: Strand-based hairstyles that are imported to Unreal Engine 4 are referred to as 'grooms'. These are imported as alembic files.
- **Niagara**: Programmable VFX system available in Unreal Engine 4, primarily used for the creation of particle simulations but is also used to simulate grooms.
- **UE4**: Abbreviation for Unreal Engine 4.

Chapter 4

Method

The scientific methods used by this study are a user experiment that utilizes the two-alternative-forced-choice (2AFC) method and a performance experiment. These are performed in order to determine the perceived realism of hair and simulation settings applied to certain hairstyles as well as measuring the performance impact of each setting. The settings are acquired through pilot experiments. The experiment design of the user experiment as well as the performance tests are motivated and presented in further detail in this chapter. The process of setting up the experiment environment is presented as well.

4.1 Pilot experiment

During the time this thesis project was started, strand-based hair rendering and simulation within UE4 was still considered an experimental feature. Therefore, in order to make sure that the experiments described in section 1.1 would be feasible on the available hardware, a few initial tests were performed. Different versions of UE4 were tested. These were version 4.24, 4.24.3, 4.25 preview 3 up until 4.25 preview 6. Tests were performed in order to determine engine stability when simulating and rendering strand-based hair, but also to subjectively evaluate the visual effect that different parameters had on the simulation and rendering. Simple, generic hairstyles which were created in Autodesk Maya were used during the pilot experiments.

The hair simulation was originally considered too unstable to experiment with, some parameter settings even resulting in engine crashes. It is unclear if crashes were caused by generic bugs or because of the hardware not reaching the necessary performance requirements needed for real-time simulation, since generic errors were thrown upon each crash. However, throughout the time period of this thesis project, the engine was being updated until it was possible to circumvent the stability issues. The experiments was therefore determined to be possible to perform.

Some initial performance tests were performed when testing the different simulation and hair settings as well. These were done with the help of the built-in GPU visualizer tool to get a rough indication of where the GPU cost of rendering a frame was going. These could only provide rough estimations and were only partially used to arrive at the simulation and hair settings which would be properly profiled later.

The quick start hair guide provided in the UE4 documentation [8] provided helpful information for how to get started with strand-based hair simulation and was generally followed throughout initial setup of the tests. Binding assets for grooms which are mentioned in the documentation, were not supported at the time and was therefore omitted. According to the documentation, this implies an added GPU cost at application startup but should not otherwise affect the performance of hair simulation.

4.2 Environment setup

In order to execute the performance experiment as well as prepare images and video for the user experiment, a formal test environment within UE4 had to be set up. The third person example map that is provided with UE4 was modified to create the environment and it was set up in version 4.25 preview 7 of Unreal Engine. The test scene contained two spotlights, one point light, one directional light, a sky box, static meshes for background and a groom asset which was parented to a skeletal mesh depicting a human female. A looped, motion-captured animation was applied to the skeletal mesh in order to provide realistic movement for the hair simulation. The animation and skinning of the mesh was provided by *Mixamo* [1]. A simple, light grey material using the default lit shading model and no textures was applied to the skeletal mesh in order to not distract from the hairstyle. The standard hair shader was applied to the groom. The environment was created to provide aesthetically pleasing lighting and to emulate what a simple game environment might look like without being too distracting or computationally demanding. Contrasting colors were chosen in order to draw attention to the character and hairstyle. Post processing was left turned on and the remaining scene settings were not modified. How the completed test environment visually looks is illustrated in Figure 4.1.

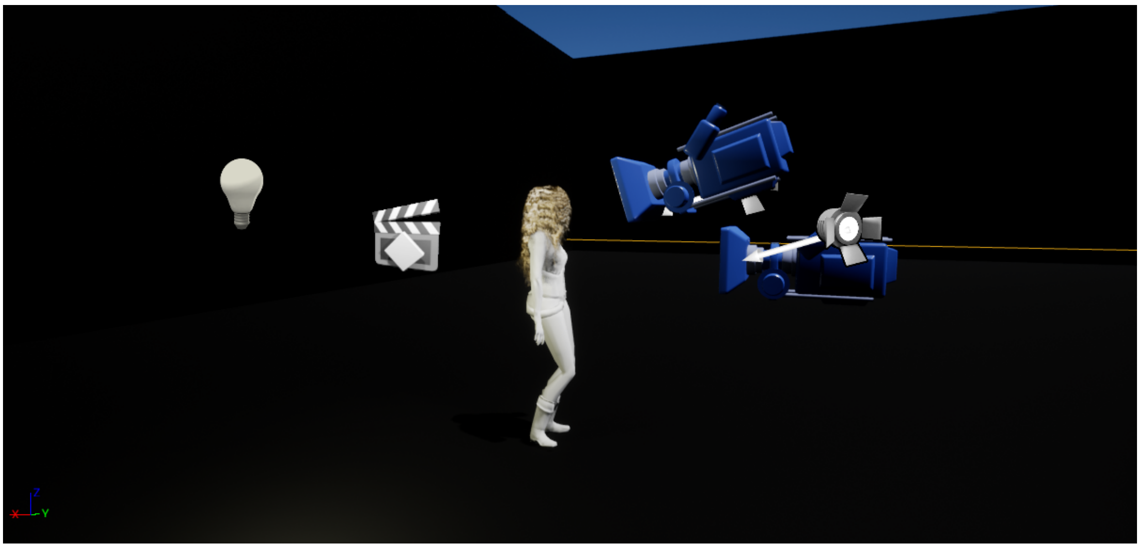

Figure 4.1: The scene used during the experiments

4.2.1 Asset creation

In order to create hairstyles for the experiment, a plugin called *Ornatrix* [11] was used within *Autodesk Maya 2019*. *XGen* within Maya and other DCC software was considered as well but in the end, Ornatrix was chosen since it provided the most straightforward user interface and tools. Seven different hairstyles with three different amount of hair strands were created. See Table 4.1 for reference. While a human head generally contains about 100 000 hair strands [22], the pilot experiments showed that simulating this amount would not be viable on the available hardware. The amount of strands was therefore decreased to 50 000 before dividing it into three different levels representing 100 percent, 60 percent and 20 percent of the full amount.

Hairstyle	Strand count	Strand thickness
Long curly (LC)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Medium curly (MC)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Short curly (SC)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Long straight (LS)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Medium straight (MS)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Short straight (SS)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05
Afro (A)	10 000	0.2
	30 000	0.2
	30 000	0.05
	50 000	0.05

Table 4.1: Created hairstyles

The hairstyles varied in length and curliness. The non-destructive workflow of Ornatrix allowed for the hair length and curliness parameters to be stacked and dynamically changed during the creation. This allowed for the creation of a base-hairstyle that could be manipulated length-wise and style-wise to create different variations. This parametric approach also helped limiting the amount of external factors which could impact the experiments and complicate synthesis of acquired results. One additional braided hairstyle was created but omitted from further tests due to not being able to be accurately physically simulated. The braid fell apart during physical simulation as illustrated by Figure 4.2. The only hairstyles which were not derived from the base hairstyle was the afro (A) and the braid. Highly curly hair such as present in the afro hairstyle can be difficult to accurately represent due to the high amount of tight curls. This requires a lot of control points per strand to be present. It was theorized that this increase in complexity may affect simulation, which is why different hair types were created.

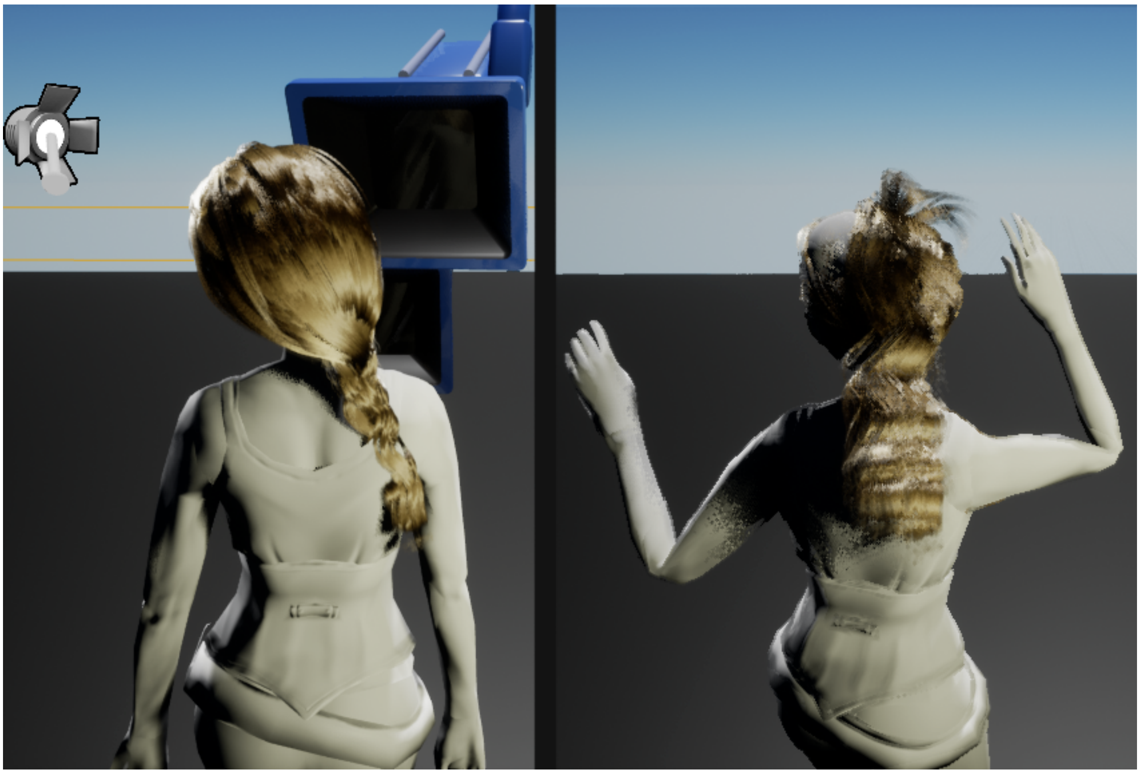

Figure 4.2: Failure case: Braid not staying together during simulation

Lastly, the hairstyles were exported into the alembic file format and imported into UE4 as groom assets. The alembic format is used by UE4 to import grooms since there is no currently defined standard file format for strand-based hair. The import settings used can be seen in Figure 4.3.

4.2.2 Simulation and hair settings

Simulation and hair rendering settings were derived from the initial pilot experiments where different parameter settings were tested in isolation. The goal when determining the hair and simulation settings to be used in the experiments, was to find settings which would have a varying effect on performance while preserving the visual result. This was especially true for the hair rendering settings which were used in the image-comparisons of the user experiment. It was theorized that in order to make up for the loss in hair coverage that a decrease in the amount of hair strands causes, the strand thickness could be somewhat increased instead. This is also stated by Yuksel and Tariq as being an easy way to increase performance [23].

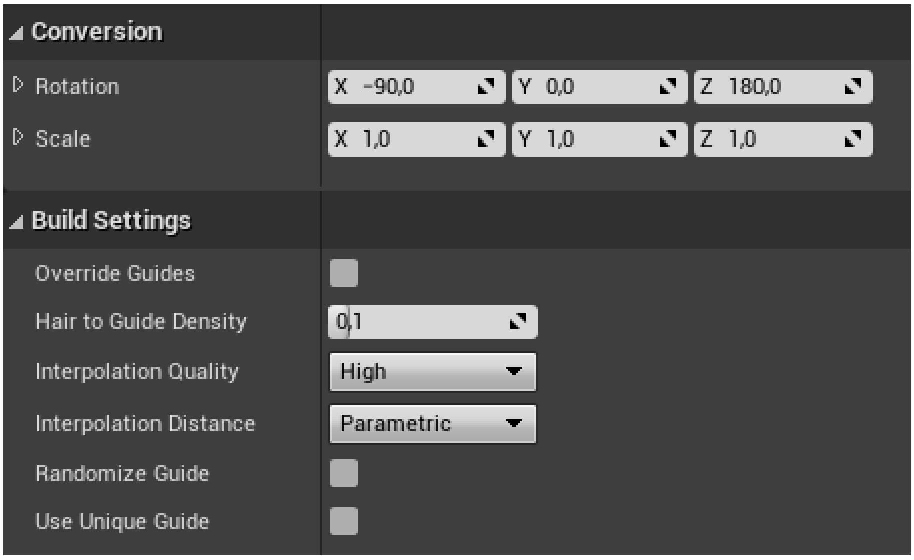

Figure 4.3: Import settings for grooms

In order to test this theory and to determine how the amount of hair strands in a hairstyle affected the perceived level of realism as well as performance, five different hair settings for the rendering of hairstyles were determined. These can be seen in Table 4.2.

Hair setting	Strand count	Strand thickness
Setting 1	10 000	0.2
Setting 2	50 000	0.05
Setting 3	30 000	0.2
Setting 4	30 000	0.05
Setting 5	10 000	0.05

Table 4.2: Hair settings

In addition to the hair settings, a different set of settings corresponding to the hair simulation were determined. When importing a groom into UE4, a set of pre-defined simulation settings are available. These settings are referred to as default settings in this thesis. The custom simulation settings were derived from these default settings and acquired by adjusting a set of parameters. The parameters were subjectively chosen according to their visual effect on hairstyles during the pilot experiments. The three different simulation settings can be seen in Table 4.3. The input values for the collision radius and bend stiffness which can be seen in the table were further adjusted with the help of a curve, which is what the words "constant" and "sigmoid curve" refers to. The different simulation settings has an impact on the physics solver and the associated constraints. More information about each attribute can be found in the hair reference guide provided in the UE4 documentation [9]. For this study, the "Angular Springs" solver was chosen.

The same default hair-shader was applied to all of the hairstyles. According to the documentation [10], this shader is based on research by d'Eon et al. [6].

Attribute	Custom setting 1	Custom setting 2	Default
Strands size	4	8	8
Strands density	6.0	3.0	1.0
Strands smoothing	0	0	0.1
Strands thickness	0.005	0.005	0.01
Strands viscosity	0.2	0.5	1.0
Collision radius	0.2 constant	[0.1, 0.001] sigmoid curve	[0.1, 0.001] sigmoid curve
Project bend	true	false	false
Project stretch	true	false	false
Grid dimensions	35	30	30
Bend stiffness	0.01 constant	[0.1, 0] sigmoid curve	[0.1, 0] sigmoid curve

Table 4.3: Simulation settings

4.2.3 Image and video rendering

Lastly, the stimuli for the user experiment was prepared. Images were rendered at 2034x1144 resolution in order to ascertain that adequate detail of the hair was preserved. All of the pictures were taken from the same camera angle and in the same lighting. The complete set of rendered images can be seen in Figure 4.4. Videos were rendered at 720p and 60 fps. Hardware limitations prevented videos from being rendered at a higher resolution which is why 720p was chosen. It was important that the rendering conditions were the same for all stimuli. Each video was exactly 10 seconds in length and taken from the same camera angle, which was slightly more zoomed-out than the camera which was used to render the still-images. An example of the final result can be seen in Figure 4.5 which is showing four frames from one of the video clips.

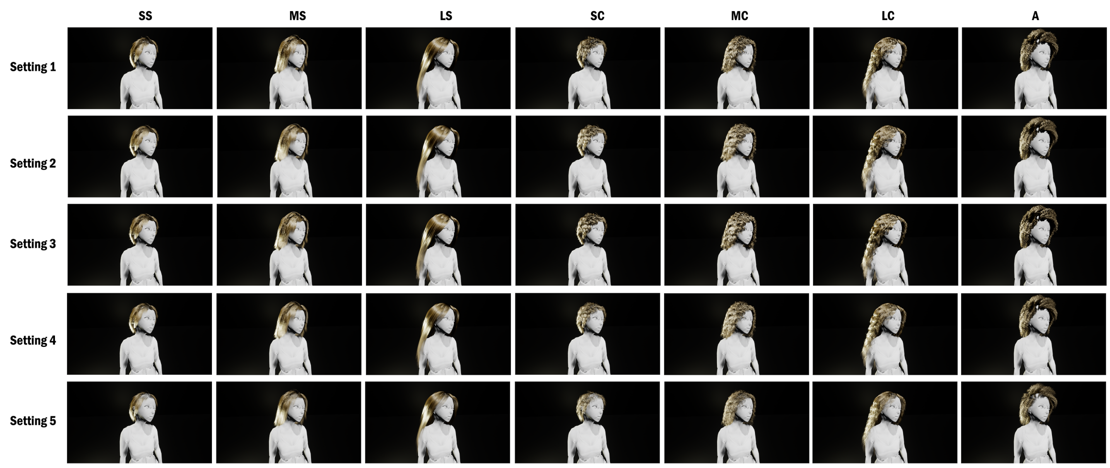

Figure 4.4: All of the images included in the user experiment

4.3 User experiment

The two-alternative-forced-choice (2AFC) method was used when designing the user experiment. Since this method is designed to give a qualitative measurement of how

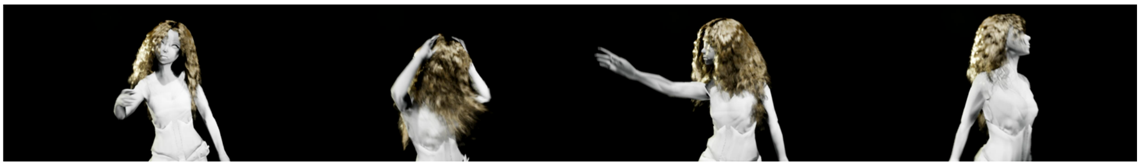

Figure 4.5: Frame sequence from one of the video clips used in the user experiment

distinct or discriminable several stimuli are from one another [5] and is commonly used to answer questions that regard visuals and perception, it was chosen for this experiment. The experiment was designed to be able to be completed online, considering the current situation which prevented the practice of performing the experiment at a physical location. Google Forms provided an easily accessible solution and was therefore chosen as the platform for the experiment. Participants were sent an invitation letter which provided a link to the Google Forms page which was created for the experiment. Instructions were provided to the participant on the first page. The participant had to consent that they had read the instructions and were at least 18 years old before continuing with participation. Participants were required to have normal or corrected-to-normal vision as well, since other target groups are outside the scope of this thesis. For details on the contents of the invitational letter see Appendix D.

During the experiment, participants were presented with stimuli in the form of pairs of still images and videos depicting the different simulation and hair settings described in section 4.2.2. Each page contained either two images or two embedded video clips which had been uploaded to YouTube. Videos on the same YouTube-channel were temporarily disabled in order for them to not show up in the recommended section at the end of playback. Participants were then asked to choose which option they perceived as the most realistic. The different settings were mixed and evenly distributed to not always be shown in the same order and in the same location to participants. These measures were taken in order to avoid participant bias. However, the image sequence could not be mixed per participant due to limitations of Google Forms. Images and video remained on the screen until the participants had chosen one of the provided options and clicked a button to proceed to the next page. Participants were asked to watch each video clip only once.

Data from the user experiment was collected between the 6th of May until the 11th of May.

4.3.1 Image- and video-pairs

28 image pairs and 21 video pairs were created. The pairs were based on comparing certain characteristics against each other since all possible combinations could not be tested. An example would be to compare a hair setting with many thin hair strands against a setting with fewer hair strands of the same thickness in order to determine the effect of decreasing the strand count. Comparing all possible combinations would have made the user experiment last longer which means a risk of introducing participant fatigue. Figure 4.6 and Figure 4.7 is illustrating which settings were compared and thus how pairs were created.

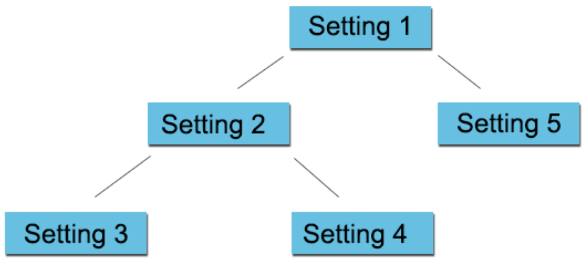

Figure 4.6: Illustration describing how hair rendering settings were compared against each other

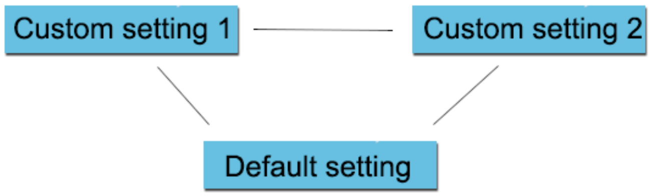

Figure 4.7: Illustration describing how simulation settings were compared against each other

4.3.2 Ethical considerations

There are always some ethical concerns when dealing with experiments that involves humans. In order for participants to not strain their eyes or get tired, the allocated time for the experiment was determined to be between 15 and 20 minutes. Participants had to be over the age of 18 in order for this experiment to follow ethical guidelines. No personal data was collected from participants and there was no obligation for participants to continue with the experiment if they changed their mind during participation. Neither were participants obligated to motivate their decision to drop out. Considering the nature of the experiment, not exposing participants to any potential dangers, there was no need for approval from the ethics committee.

4.4 Performance profiling

By the time that performance profiling could be performed, version 4.25 of Unreal Engine had been officially released which is why this version of the engine was used during the performance experiment. In order to determine the general performance of the different simulation settings, the built in profiling tools in UE4 were used. These tools are able to provide detailed statistics about the execution times and CPU and GPU performance as well as an average frame-rate derived from recorded play-sessions. In this case, the CSV Profiler was used rather than the session frontend profiler. The CSV profiler was chosen in order to create graphs from the acquired data, but also because of its' ability to capture a set amount of frames. This would ease the process of analyzing the data later on. In this case, 60 frames were sampled. To ensure that there would be as little noise in the data as possible, there were no other GPU workloads during the profiling process and V-sync was turned off. During the performance tests, the same scene was used as for the user experiment.

Hair of different strand counts as well as the different simulation settings were profiled. In order to simplify comparison between the different simulation settings, the different hairstyles were all simulated with 30 000 hair strands and when profiling the strand counts, the default simulation setting was applied to all hairstyles. The same strand thickness was used during all performance tests. Graphs were created to show the execution time of each frame, GPU total, the game thread, the render thread and Niagara GPU simulation.

4.4.1 Enabled console commands

The following console commands were input before starting the profiling session in order to acquire accurate profiling data via the CSV profiler in UE4:

- r.GPUCsvStatsEnabled 1
- r.VSync 0

Chapter 5

Results and Analysis

This chapter presents and analyzes the results of the user experiment as well as the performance experiment.

5.1 User experiment results

The user experiment had a total of 18 participants. Each participant could vote once per image- or video-pair. The results have been visualised into charts according to the different compared settings. The vote distribution on image pairs can be seen in Figure 5.1 while the vote distribution on video pairs can be seen in Figure 5.2. An overview of all the votes on all of the pairs can be found in Appendix C.

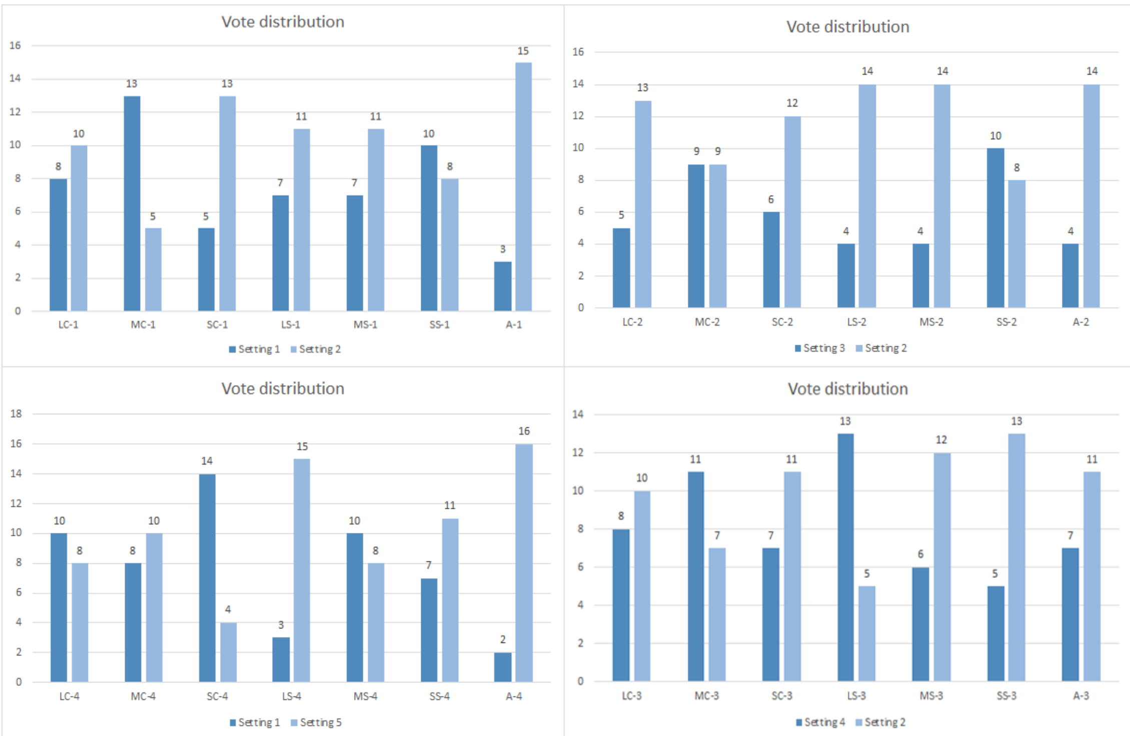

Figure 5.1: Vote distribution on image-pairs

In order to determine if a significant difference was present in the voting, an online tool [20] was used to perform a chi-square test on each pair. The difference is presented in Table 5.1 and 5.2 as a p-value. A significant difference could be

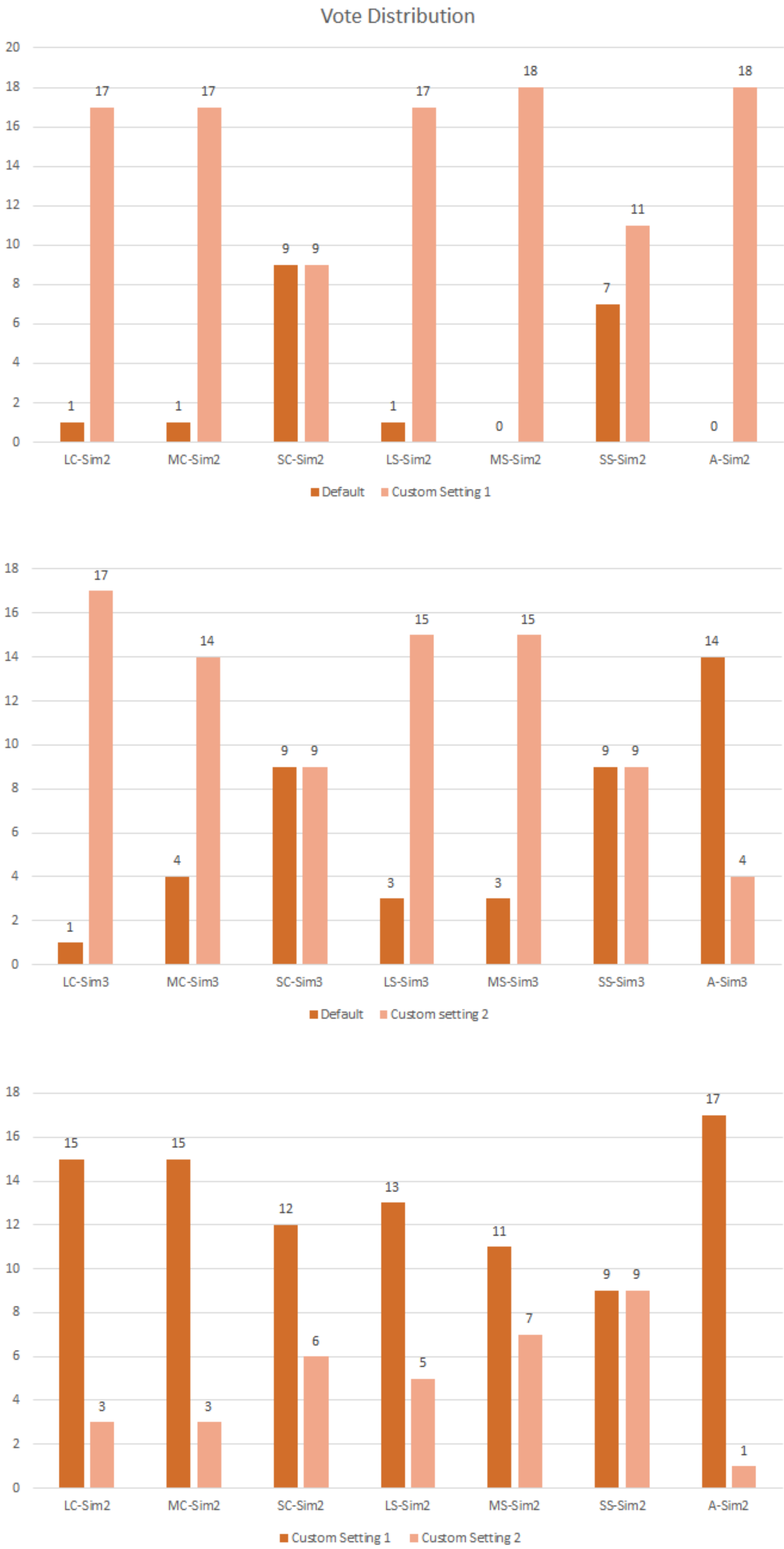

Figure 5.2: Vote distribution on video-pairs

determined if the p-value was calculated to be smaller than 0.05, which implies a very small chance of votes being randomly assigned. The observed frequency was derived from the amount of votes each option received while the expected frequency was derived from the amount of participants divided by two.

Pair ID	p-value	Compared A	Compared B
MC-2	1	Setting 3	Setting 2
LC-1	0.63735189	Setting 1	Setting 2
SS-1	0.63735189	Setting 1	Setting 2
SS-2	0.63735189	Setting 3	Setting 2
LC-3	0.63735189	Setting 4	Setting 2
LC-4	0.63735189	Setting 1	Setting 5
MC-4	0.63735189	Setting 1	Setting 5
MS-4	0.63735189	Setting 1	Setting 5
LS-1	0.34577859	Setting 1	Setting 2
MS-1	0.34577859	Setting 1	Setting 2
MC-3	0.34577859	Setting 4	Setting 2
SC-3	0.34577859	Setting 4	Setting 2
A-3	0.34577859	Setting 4	Setting 2
SS-4	0.34577859	Setting 1	Setting 5
SC-2	0.15729921	Setting 3	Setting 2
MS-3	0.15729921	Setting 4	Setting 2
MC-1	0.05934644	Setting 1	Setting 2
SC-1	0.05934644	Setting 1	Setting 2
LC-2	0.05934644	Setting 3	Setting 2
LS-3	0.05934644	Setting 4	Setting 2
SS-3	0.05934644	Setting 4	Setting 2
LS-2	**0.01842213**	Setting 3	Setting 2
MS-2	**0.01842213**	Setting 3	Setting 2
A-2	**0.01842213**	Setting 3	Setting 2
SC-4	**0.01842213**	Setting 1	Setting 5
A-1	**0.00467773**	Setting 1	Setting 2
LS-4	**0.00467773**	Setting 1	Setting 5
A-4	**0.00096743**	Setting 1	Setting 5

Table 5.1: Still-image evaluation results sorted by p-value

Results show that there was generally a greater perceived difference between the simulation settings used in the video-pairs than in the hair settings used in the image-pairs. It was also more common for settings to be perceived differently when applied to longer hair rather than short hair with no regard to pairs being images or video.

The results of the voting on the image-pairs shows that thin hair strands were generally preferred over thicker hair strands except for in a few cases when applied to hairstyles which consisted of only 10 000 strands of hair. When comparing hairstyles which consisted of 30 000 strands and 50 000 strands of the same strand thickness, no significant difference between the settings could be noticed by the participants except for in two cases. These cases were the long straight hairstyle (LS) where a

Pair ID	p-value	Compared A	Compared B
SC-Sim1	1	Default	Custom setting 1
SC-Sim3	1	Default	Custom setting 2
SS-Sim2	1	Custom setting 1	Custom setting 2
SS-Sim3	1	Default	Custom setting 2
SS-Sim1	0.34577859	Default	Custom setting 1
MS-Sim2	0.34577859	Custom setting 1	Custom setting 2
SC-Sim2	0.15729921	Custom setting 1	Custom setting 2
LS-Sim2	0.05934644	Custom setting 1	Custom setting 2
MC-Sim3	**0.01842213**	Default	Custom setting 2
A-Sim3	**0.01842213**	Default	Custom setting 2
LC-Sim2	**0.00467773**	Custom setting 1	Custom setting 2
MC-Sim2	**0.00467773**	Custom setting 1	Custom setting 2
LS-Sim3	**0.00467773**	Default	Custom setting 2
MS-Sim3	**0.00467773**	Default	Custom setting 2
LC-Sim1	**0.00016244**	Default	Custom setting 1
LC-Sim3	**0.00016244**	Default	Custom setting 2
MC-Sim1	**0.00016244**	Default	Custom setting 1
LS-Sim1	**0.00016244**	Default	Custom setting 1
A-Sim2	**0.00016244**	Custom setting 1	Custom setting 2
MS-Sim1	**0.00002209**	Default	Custom setting 1
A-Sim1	**0.00002209**	Default	Custom setting 1

Table 5.2: Video evaluation results sorted by p-value

decreased hair count was preferred and the short straight hairstyle (SS) where an increased hair count was preferred. There was generally a preference for setting 1 which consisted of 50 000 hair strands of size 0.05 but in most cases, a significant difference between using many and fewer hair strands was not found, even in extreme cases when comparing hairstyles with 10 000 strands to hairstyles with 50 000 strands.

The vote distribution on the video pairs show that both custom setting 1 and custom setting 2 were preferred over the default simulation setting in all cases but for very short hair. Out of the custom simulation settings, simulation setting 1 was predominantly perceived as the most realistic in the case of longer and curlier hair. The results suggest that the shorter the hair, the more difficult it became for participants to tell the settings apart.

5.2 Performance results

Performance profiling was done on a desktop PC, running 64-bit Windows 10 Pro N. The PC is equipped with an Intel core i7-4790k processor, an NVIDIA GeForce GTX 970 graphics card and 16 gigabytes of RAM. Unreal Engine version 4.25 was used during the performance measurements.

The measured execution times seen in Table 5.3 and 5.4 have been averaged over 60 frames and are presented in milliseconds. The results from the performance tests show that there was a relatively small difference between each simulation setting, but also that Custom setting 1 was slightly more computationally expensive than the other settings. Interesting to note is that Niagara GPU simulation, which drives the solver of the hair simulation, generally shows to be unaffected by the difference in parameters when comparing the performance of the different simulation settings. The style of the hair did not have a significant impact on the performance results. A significant difference in performance could however be seen when comparing different strand counts, with more hair strands having to be rendered and simulated resulting in much longer execution and rendering times. Performance differences can also be seen when comparing the results of hair of different length to each other. The results show that the shorter the hair, the better performance was achieved. For detailed graphs, the reader is referred to Appendix A and B. It should also be noted that very few measurements of the frame-time were below the threshold of 16.6 ms. This threshold commonly indicates a frame rate of 60 frames per second, which is optimal when it comes to video games. However, results all showed frame-times under 33.3 ms, indicating that a stable 30 frames per second is possible to acquire for most of the hairstyles. This shows that performance was still within the current state-of-the-art standards for frame rates in video games, which lies between 30-60 frames per second.

Style	Length	Simulation setting	Frame	Total GPU	Game Thread	Render Thread	Niagara GPU Simulation
Curly	Long	Default	26.69	26.51	4.60	1.00	1.36
Curly	Long	Custom setting 1	30.45	30.46	4.30	1.00	1.02
Curly	Long	Custom setting 2	29.43	29.26	4.68	1.14	1.38
Curly	Medium	Default	18.49	18.43	4.31	0.98	1.26
Curly	Medium	Custom setting 1	21.56	21.47	0.17	8.61	0.87
Curly	Medium	Custom setting 2	21.42	21.39	4.34	1.00	1.23
Curly	Short	Default	15.01	11.47	4.23	0.98	1.11
Curly	Short	Custom setting 1	16.25	16.13	4.23	7.93	0.81
Curly	Short	Custom setting 2	14.86	14.81	4.50	1.09	1.10
Straight	Long	Default	23.75	23.78	4.55	0.99	1.36
Straight	Long	Custom setting 1	26.12	26.05	4.41	0.01	1.00
Straight	Long	Custom setting 2	23.79	23.77	4.49	0.99	1.36
Straight	Medium	Default	19.26	19.22	4.93	0.99	1.24
Straight	Medium	Custom setting 1	22.55	22.42	0.17	8.51	0.89
Straight	Medium	Custom setting 2	21.78	21.74	5.04	0.98	1.27
Straight	Short	Default	15.10	15.07	4.16	0.98	1.12
Straight	Short	Custom setting 1	15.63	15.55	0.17	7.98	0.82
Straight	Short	Custom setting 2	14.75	14.65	4.47	1.09	1.13
Afro	-	Default	24.06	23.58	0.23	11.79	1.11
Afro	-	Custom setting 1	24.24	24.14	0.17	8.27	0.76
Afro	-	Custom setting 2	26.24	26.09	0.11	9.21	1.15

Table 5.3: Performance measurements for different hair simulation settings (ms)

Style	**Length**	**Strand count**	**Frame**	**Total GPU**	**Game Thread**	**Render Thread**	**Niagara GPU Simulation**
Curly	Long	10 000	18.09	18.06	3.63	1.03	0.70
Curly	Long	30 000	26.69	26.51	4.60	1.00	1.36
Curly	Long	50 000	33.04	33.00	5.48	1.04	1.99
Curly	Medium	10 000	15.27	15.24	3.71	0.99	0.66
Curly	Medium	30 000	18.49	18.43	4.31	0.98	1.26
Curly	Medium	50 000	27.49	27.47	6.34	0.99	1.78
Curly	Short	10 000	11.53	14.95	3.53	1.00	0.61
Curly	Short	30 000	15.01	11.47	4.23	0.98	1.11
Curly	Short	50 000	19.25	19.20	5.00	1.05	1.62
Straight	Long	10 000	19.77	19.67	3.67	1.02	0.69
Straight	Long	30 000	23.75	23.78	4.55	0.99	1.36
Straight	Long	50 000	31.61	31.72	5.63	1.02	2.01
Straight	Medium	10 000	15.48	15.38	3.71	0.97	0.66
Straight	Medium	30 000	19.26	19.22	4.93	0.99	1.24
Straight	Medium	50 000	25.66	25.50	0.22	12.04	1.74
Straight	Short	10 000	12.05	11.97	3.68	1.03	0.62
Straight	Short	30 000	15.10	15.07	4.16	0.98	1.12
Straight	Short	50 000	18.33	18.27	5.01	1.04	1.61
Afro	-	10 000	17.28	17.23	4.11	1.00	0.66
Afro	-	30 000	24.06	23.58	0.23	11.79	1.11
Afro	-	50 000	27.01	27.62	7.15	1.66	1.62

Table 5.4: Performance measurements for different hair strand counts (ms)

Chapter 6

Discussion

A user experiment and a performance experiment were performed in order to collect data on the perceived realism and computer performance of different simulation settings applied to different types of strand-based hairstyles. The data was successfully collected and synthesized in order to answer the asked research questions of this thesis. This chapter discusses the results and findings of the user experiment as well as the performance tests. Lastly the viability of the achieved results will be discussed.

6.1 Visual impact of customized settings

The results of the user experiment were not entirely expected. The vote distribution on image-pairs suggests that participants in general found it difficult to discern differences in the images. Evenly distributed votes in a 2AFC task can be a sign of uncertainty, as described by Cunningham [5]. Perhaps the changes in each hair setting were too subtle for participants to notice, or perhaps the viewing angle did not provide enough detail of each hairstyle, resulting in uncertainty and random voting. This made it difficult to analyze and determine relations between the data since barely any significant difference could be found in the voting. What could be determined was a preference for finer hair strands, regardless of hair strand count, which was unexpected. In all of the significant cases, this preference can be seen. Expanding the hair strand thickness was supposed to make up for the volume loss that a reduction in hair strand count introduced. Fine hair strands were however, still generally perceived as more realistic than the thicker hair strands, which should not be surprising considering hair strands in real life are extremely thin, only being 0.1 millimeter in diameter [22]. Contrary to initial theories, this suggests that hair coverage did not count as a significant factor in the perceived realism of a hairstyle, at least when the hair is not in motion, or too short.

It has been shown in earlier studies that light and shadow has an impact on what is perceived as realistic and not [21, 18]. It is possible that the hair shading could therefore have impacted the perceived level of realism during the experiment as well. This should be considered when utilizing the results of this study in future work. If another shader than the standard hair shader in UE4 is used, the results of the user experiment are not guaranteed to be as trustworthy.

It was hypothesized that the application of different customized settings would heighten the perceived level of realism in different types of hairstyles. The results show that there is reason to believe that this is true since there was a significant preference seen for the customized simulation settings in the majority of the hairstyles, especially for simulation setting 1. This was the simulation setting which deviated the most from the default simulation setting. It could be determined that shorter hairstyles do not profit significantly from the application of customized simulation settings since participants generally did not show a preference in these cases. Differences in simulation settings applied to short hairstyles could not be easily detected by the participants of the user experiment. This suggests that computationally cheaper simulation methods can be applied to short hairstyles in order to increase performance without sacrificing visual fidelity. It is more important to consider the visual impact of physical simulation in the case of longer hair, since the impact of the simulation proved to be more obvious to participants in these cases.

There are many properties which may impact the physical behaviour and the overall perceived level of realism in different types of hair. The complexity of hair can make it very difficult to determine what looks visually correct and not. It is possible that expectations of what different hairstyles should look like might vary between different people which made the method of performing a perceptual experiment relevant for this study. The pilot experiment which was performed in order to determine which parameter settings should be used for the experiment would have profited from testing individual parameters in a similar manner. This was deliberately not done due to time limitations.

As discussed by Jung and Lee [13], it is important that many different types of hairstyles are tested in a study like this. This is a reason why seven different hairstyles were chosen and an additional braid was created. It was however, determined from the pilot experiment in this study that constrained hairstyles such as the braid could not be accurately simulated. This is mainly due to the way that the hair simulation system and strand interpolation works in UE4. One proven solution to this issue would be to categorize parts of the hairstyle and apply certain simulation settings to each part [13]. This could be done either with artistic intervention or by automatic methods as in Jung and Lee's work, but would require an entirely different kind of system than the current implementation. Other solutions include skinning the braid to a bone, apply physics simulation to the bone and simply have the braid follow the bone instead of simulating the guide hairs of the hairstyle. Although this means sacrificing the ability of individual strands to be simulated, thus making the hair less realistic. These methods were not explored since this would increase the scope of this thesis, but would have been interesting to explore nonetheless. An attempt at creating a natural-looking afro was made as well, since natural African hair was one of the unexplored areas of Jung and Lee's work. However, the preferred settings for this hairstyle was not found to differentiate significantly from the other styles. It was the only hairstyle where the default simulation setting was preferred over simulation setting 2, even though simulation setting 1 was still the most preferred setting though. Since individual parameters were not tested, the reason for this somewhat contradicting phenomena cannot be fully explained. Perhaps a reason could be that it was the first video-pair which was shown to participants, or the unique parameter values for the strands density or strands viscosity could be a factor. If more hairstyles of the same hair type had been included in the study, it would have been easier to determine if the phenomena was caused by deliberate choice or some other factor.

6.2 Performance impact of customized settings

The current standards in the video game industry generally demands games to be run at between 30 to 60 frames per second. The results of the performance experiment showed that while execution and rendering times rarely were below the threshold of 16.6 ms, a relatively stable framerate would still be possible to acquire, especially when simulating fewer hair strands. Since the simulation performance is highly dependant on which underlying hardware it is being run on, as stated in the Unreal Engine 4 documentation [7], it is believed that a more recent or powerful graphics card could provide an adequate performance boost. The factor which had the biggest impact on performance was shown to be the amount of hair strands that a hairstyle contains. This confirms that using hair strand reduction together with an increase in strand thickness could be used as a form of LOD-optimization, as described in [23] but when considering the results from the user experiment it also becomes apparent why this could be a bad idea. The results of the user experiment emphasizes the importance of hair strands remaining fine in order to appear realistic. Therefore, perhaps other solutions that decreases the amount of hair strands without increasing the strand thickness should be explored, such as the methods described by Jansson et al. [12].

The customized simulation settings did not have the expected impact on computer performance. The impact on performance of the simulation settings was not very noticeable compared to the effects of the hair strand count. The reason for this is likely to be due to which parameters were chosen for the experiments. It is believed that the parameters which were chosen did not necessarily change the execution of the physical simulation significantly. Perhaps this is due to the parameters being related to the same physics solver. With this said, there is currently very little information available on the execution of the strand-based hair simulation in UE4. Position-based-dynamics is already a highly efficient simulation method [3]. It is likely that the extended version [15] is being used for hair simulation in UE4 as the method is mentioned in the documentation [9]. The results make is quite apparent that hair strand simulation in UE4 is bound by the GPU rather than the CPU. In order for many hair strands to be simulated simultaneously, the simulation is most probably GPU-driven, which would be highly efficient for performance as well.

The results therefore suggest that it is better to choose simulation settings which looks visually pleasing and focus on decreasing the strand count of hairstyles in order to increase computer performance. However, it is possible that the values which were chosen simply were not drastic enough for the change in performance to be noticeable or that the data contains noise, even though adequate measures were taken to avoid this. It is also possible that "Niagara GPU Simulation" does not reflect the performance of the physics solver entirely. When looking at the frame time though, the change in performance between simulation settings is still minimal.

6.3 Viability of results

The biggest validity threat to this study is that the environment of the user experiment was not completely controlled. The user experiment had to be performed online due to the current situation regarding the covid-19 outbreak. Therefore, there are many potential external factors which could not be controlled. Participants may for example have different screen size, resolution, screen brightness and internet connection speed. In the instructions for the experiment it said that participants should watch each video only once and at the highest resolution, but neither is it guaranteed that all participants followed the instructions. Additionally, the experiment was not timed, which allowed participants to watch each image pair for as long as they would like before choosing an option. The time spent on the experiment might therefore vary from one participant to another.

All participants received the images and video pairs in the same predetermined order. The order was originally planned to be randomized for each participant in order to minimize the chance of participant bias but due to being limited by the use of Google Forms, this could not be done.

An option which was considered in order to mitigate these issues, was to develop or use a small program which participants could download. In that way, the time for each stimuli to be visible on the screen could be controlled as well as randomization for the order of the stimuli. Developing a program like this was however, outside the scope of this thesis and there would still have been some uncontrollable external factors present.

There were also only 18 participants in the user study. While the amount of participants can be considered adequate for a small study like this, a greater amount of participants could have provided more reliable results. In addition to the 2AFC questions, it would also have been interesting to ask participants more about their experience via further questioning. One alternative could have been to use a similar scale as seen in [21] where participants could rank their opinion of different perceived attributes in each image they were shown from 1 to 5. This would have provided more insight into why participants chose to vote in a certain way.

Due to time restrictions for the experiment, the amount of images and settings that could be compared had to be limited. Comparing each setting against each other in a formal experiment would have made more sense and given the possibility of gaining more accurate results as well. This could be an area to explore in future work though.

Lastly, it is important to note that the results of this study are only viable in the environment of UE4.

Chapter 7

Conclusions and Future Work

The conclusions of this work are presented in this chapter, as well as suggestions for future work within the topic of this thesis.

7.1 Conclusions

The area of real-time strand-based hair rendering and simulation was explored in this thesis in order to provide an insight into what can be done to increase the visual appeal and computer performance of physical simulation for different types of strand-based hairstyles.

A set of objectives were declared and two research questions were raised in order to achieve the aim of this thesis. The objectives of the study were successfully completed and data could be gathered and synthesized from the experiments in order to answer the research questions.

To determine what simulation settings inside Unreal Engine 4 produces the most realistic results for a set of different types of strand-based hairstyles, a pilot experiment had to be performed in addition to the user experiment. Since certain limitations on time had to be considered for the study as well as the experiment, a set of hair and simulation settings were first determined by the author of this thesis.

Out of these settings, the simulation settings which produces the most realistic results could be determined to be the customized simulation settings, in particular custom setting 1, which was also determined to be slightly more computationally expensive than the other settings. A strong preference for the custom settings was found in all hairstyles but those of very short hair length. Therefore it can be concluded that longer hairstyles benefit more from having customized simulation settings applied to them in order to be perceived as more realistic. The specific combinations of parameter settings which were used in the experiment provided a heightened perceived level of realism on the tested hairstyles. An additional finding from the experiments was that constrained hairstyles could not be simulated in a realistic manner with the methods used in this thesis.

Which specific attributes has the most impact on the perceived realism could not be clearly determined due to the way that testing was performed. The effect of each attribute was only briefly tested during the pilot experiment. What could be concluded was that hair strand thickness is one of the most important factors for creating realistic-looking hair, as shown by the user experiment. There is however a slight possibility that the preference for fine hair strands was caused by some external factor, since it was found that participants of the user experiment generally had a difficult time distinguishing between the images used.

In order to optimize computer performance without sacrificing visual fidelity, it can be concluded that it is important to be aware of the amount of hair strands a specific hairstyle consists of. While simulation settings were determined to not affect computer performance greatly, the hair strand count was determined to be a major contributing factor. The results show that lowering the strand count for certain hairstyles, especially for long hair, could provide a performance increase without necessarily experiencing a noticeable decrease in perceived realism. Short hair on the other hand can be difficult to accurately represent with a lower amount of hair strands, but even then it is important that strands appear fine enough to be considered hair-like.

The hypothesis regarding the possibility of achieving better computer performance and a heightened perceived level of realism by the application of customized simulation settings to certain types of hairstyles can be concluded as partially proven right. While the perceived level of realism benefited greatly from the application of customized settings, computer performance was not greatly affected. Performance was more affected by the amount of hair strands which had to be simulated. Thus, the hair rendering settings are more important to consider when trying to optimize strand-based hair simulation. It can therefore be concluded that the results of this thesis could be used to achieve a limited performance boost.

The experiments performed in this study found that the finer details of hair has a bigger impact on the overall perceived realism of a hairstyle. This suggests that there is a strong reason to keep investigating strand-based approaches for real-time hair simulation when pursuing photo-realism, as this may currently provide the most detailed hair simulation. As hardware becomes increasingly capable of handling strand-based hair rendering and simulation, the experiments performed in this thesis shows that strand-based hair has a chance of becoming a viable alternative to using card-based hairstyles in video games in the future. This would help in bringing the realism in digital humans to a new level of fidelity and thus help players feel more immersed in the stories which are told through the video game medium.

7.2 Future work

There are many areas of this work which could be further explored. Since real-time hair rendering is still in the early stages of development, future work should first and foremost focus on stability improvements for hair simulation. Since performance tests were done on only one hardware setup, it would be interesting to see how well hair simulation runs on different or more recent hardware. Additionally, in order to further increase computer performance, it is believed that LOD-methods could be explored with favorable results.

One of the biggest shortcomings of this thesis is that it could not be clearly determined how all of the individual parameters used in the experiments affected the results. A subjective evaluation of each of the attributes was performed during the pilot experiment in this thesis, but this is not discussed in detail since very little technical knowledge was found. It would therefore be beneficial to test these parameters in isolation to investigate their effects further. Additionally, performing the experiments with a bigger amount of participants would be of interest in order to increase the accuracy of the results.

While the created hairstyles included in this study provided some variation, there are still some unexplored styles which would be interesting to explore in future work. Since braided hair could not be simulated via the methods presented in this thesis, it would be interesting to explore other approaches that could accurately simulate constrained hairstyles such as braids and hair buns for example.

Since UE4 provides the possibility of creating entirely new physics solvers, it would be interesting to see if it would be possible to implement a customized solver for strand-based hair or even particular hairstyles. It is also of interest to compare the performance of different physics solvers since the performance tests in this thesis could not determine a significant difference between the tested simulation settings.

References

[1] Adobe. Mixamo, 2020. `https://www.mixamo.com/#/`.

[2] Tobias Andersen, Viggo Falster, Jeppe Frisvad, and Niels Christensen. Hybrid fur rendering: combining volumetric fur with explicit hair strands. *The Visual Computer*, 32, 05 2016.

[3] Jan Bender, Matthias Müller, and Miles Macklin. A survey on position based dynamics, 2017. In *Proceedings of the European Association for Computer Graphics: Tutorials*, EG '17, Goslar, DEU, 2017. Eurographics Association.

[4] Menglei Chai, Changxi Zheng, and Kun Zhou. Adaptive skinning for interactive hair-solid simulation. *IEEE Transactions on Visualization and Computer Graphics*, 23:1–1, 04 2016.

[5] Douglas W. Cunningham and Christian Wallraven. *Experimental Design : From User Studies to Psychophysics*. CRC Press LLC, Florida, 2011.

[6] Eugene d'Eon, Steve Marschner, and Johannes Hanika. Importance sampling for physically-based hair fiber models. In *SIGGRAPH Asia 2013 Technical Briefs*, SA '13, New York, NY, USA, 2013. Association for Computing Machinery.

[7] Epic Games. Hair rendering and simulation overview, 2020. `https://docs.unrealengine.com/en-US/Engine/HairRendering/Overview/index.html` Accessed: 2020-05-18.

[8] Epic Games. Hair Rendering and Simulation Quick Start, 2020. `https://docs.unrealengine.com/en-US/Engine/HairRendering/QuickStart/index.html` Accessed: 2020-05-22.

[9] Epic Games. Hair rendering and simulation settings and properties, 2020. `https://docs.unrealengine.com/en-US/Engine/HairRendering/Reference/index.html` Accessed: 2020-05-22.

[10] Epic Games. Photorealistic Character, 2020. `https://docs.unrealengine.com/en-US/Resources/Showcases/PhotorealisticCharacter/index.html` Accessed: 2020-06-08.

[11] Ephere Inc. Ornatrix for maya, 2020. `https://ephere.com/plugins/autodesk/maya/ornatrix/` Accessed: 2020-05-12.

[12] Erik Sven Vasconcelos Jansson, Matthäus G. Chajdas, Jason Lacroix, and Ingemar Ragnemalm. Real-Time Hybrid Hair Rendering. In *Eurographics Symposium on Rendering - DL-only and Industry Track*, 2019.

[13] Seunghwan Jung and Sung-Hee Lee. Hair modeling and simulation by style. *Computer Graphics Forum*, 37(2):355–363, 2018.

[14] Jerome Lengyel, Emil Praun, Adam Finkelstein, and Hugues Hoppe. Real-time fur over arbitrary surfaces. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, page 227–232, New York, NY, USA, 2001. Association for Computing Machinery.

[15] Miles Macklin, Matthias Müller, and Nuttapong Chentanez. Xpbd: Position-based simulation of compliant constrained dynamics. In *Proceedings of the 9th International Conference on Motion in Games*, MIG '16, page 49–54, New York, NY, USA, 2016. Association for Computing Machinery.

[16] Nadia Magnenat-Thalmann and Daniel Thalmann. Virtual humans: thirty years of research, what next? *The Visual Computer*, 21:997–1015, 2005.

[17] Timothy Martin, Wolfgang Engel, Nicolas Thibieros, Jason Yang, and Jason Lacroix. Tressfx: Advanced real-time hair rendering. In *GPU Pro 5: Advanced Rendering Techniques*, chapter 5, pages 193–209. CRC Press, 2014.

[18] Rachel McDonnell, Martin Breidt, and Heinrich Bülthoff. Render me real? investigating the effect of render style on the perception of animated virtual humans. *ACM Transactions on Graphics*, 31:91: 1–11, 07 2012.

[19] Matthias Müller, Tae Kim, and Nuttapong Chentanez. Fast simulation of inextensible hair and fur. 12 2012.

[20] K. J. Preacher. Calculation for the chi-square test: An interactive calculation tool for chi-square tests of goodness of fit and independence, 2001. `http://quantpsy.org` Accessed: 2020-05-14.

[21] Girish Ramesh, Martin Turner, Björn Schröder, and Franz Wortmann. Analysis of hair shine using rendering and subjective evaluation. *ACM Trans. Appl. Percept.*, 15(4), October 2018.

[22] Kelly Ward, Florence Bertails, Tae-Yong Kim, Stephen R. Marschner, Marie-Paule Cani, and Ming C. Lin. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):213–234, March 2007.

[23] Cem Yuksel and Sarah Tariq. Advanced techniques in real-time hair rendering and simulation. In *ACM SIGGRAPH 2010 Courses*, SIGGRAPH 2010, pages 1:1–1:168, New York, NY, USA, 2010. ACM.

[24] Aron Zoellner. NVIDIA HairWorks in the Witcher 3, 2015. `https://developer.nvidia.com/content/nvidia-hairworks-witcher-3` Accessed: 2020-05-17.

Appendix A

Performance data for simulation settings

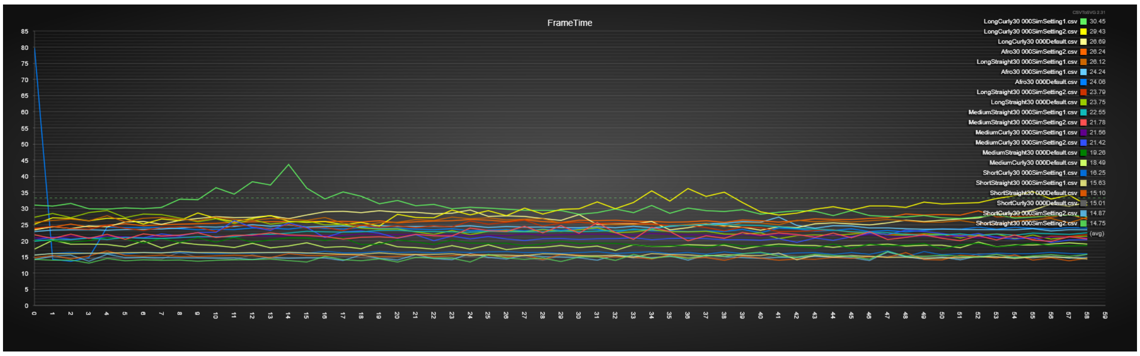

Figure A.1: Frame time (ms)

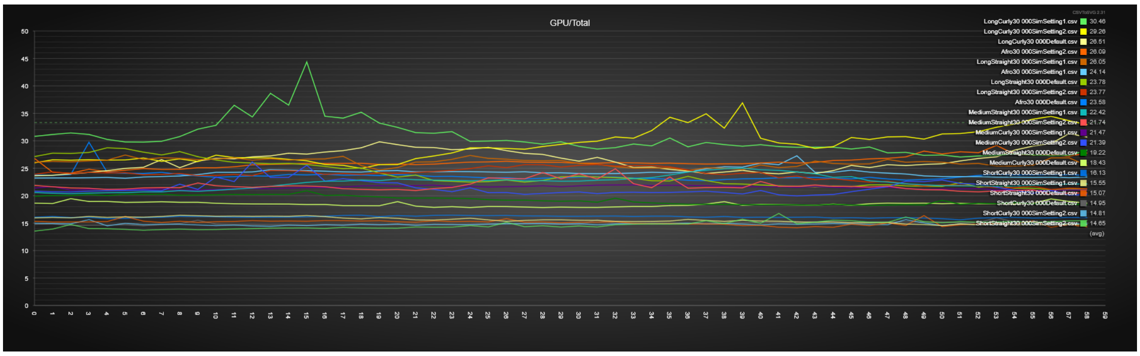

Figure A.2: GPU total time (ms)

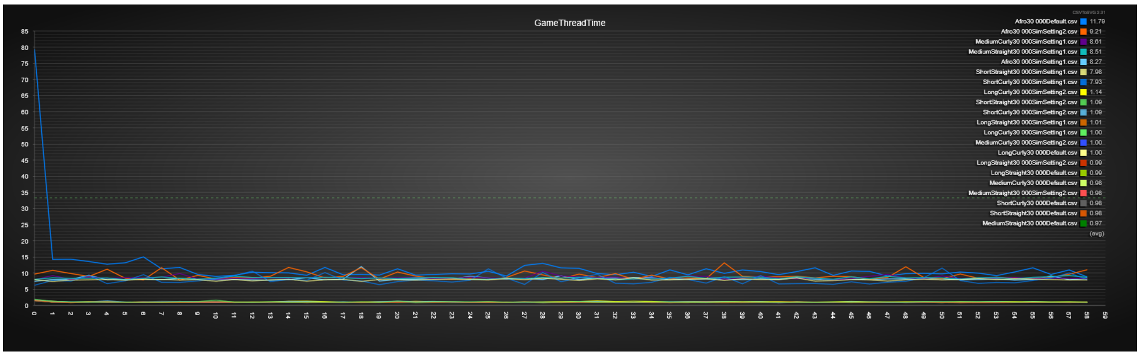

Figure A.3: Game thread time (ms)

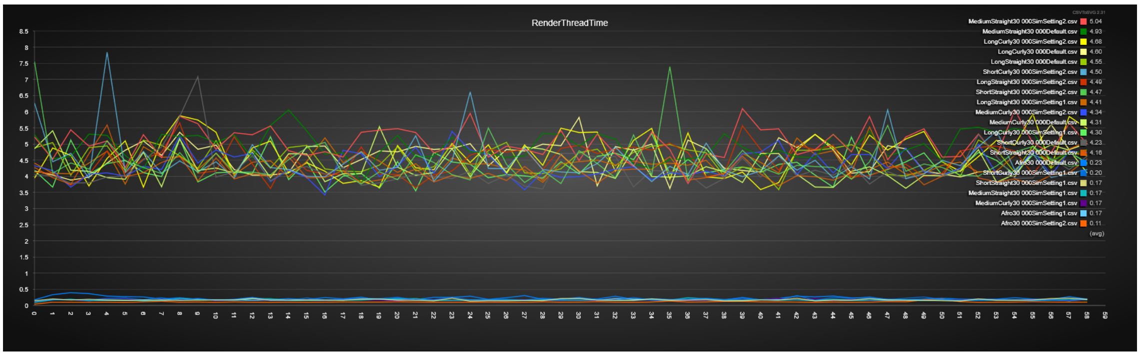

Figure A.4: Render thread time (ms)

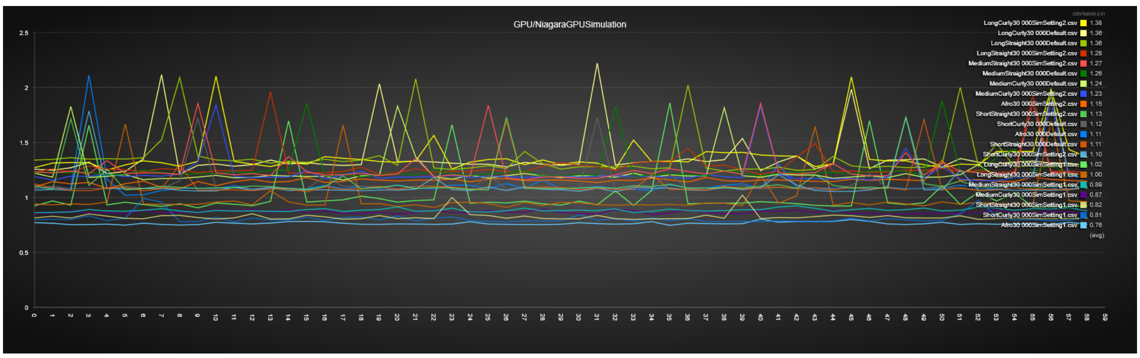

Figure A.5: Niagara GPU simulation time (ms)

Appendix B

Performance data on different hairstyle strand count

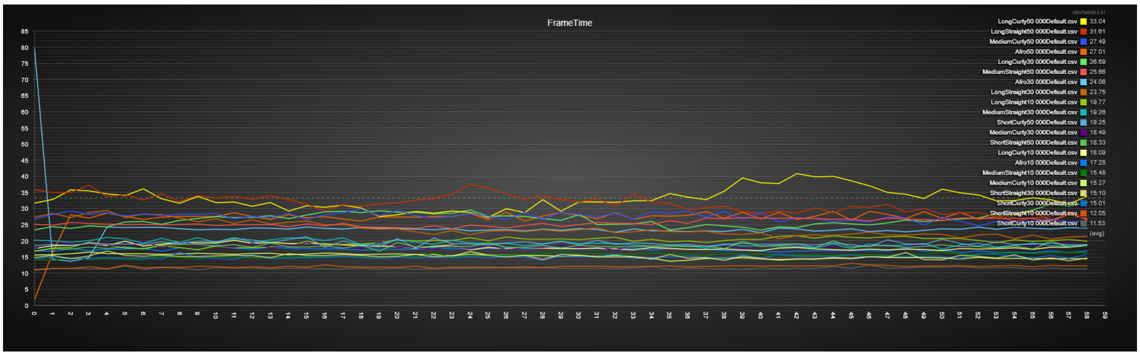

Figure B.1: Frame time (ms)

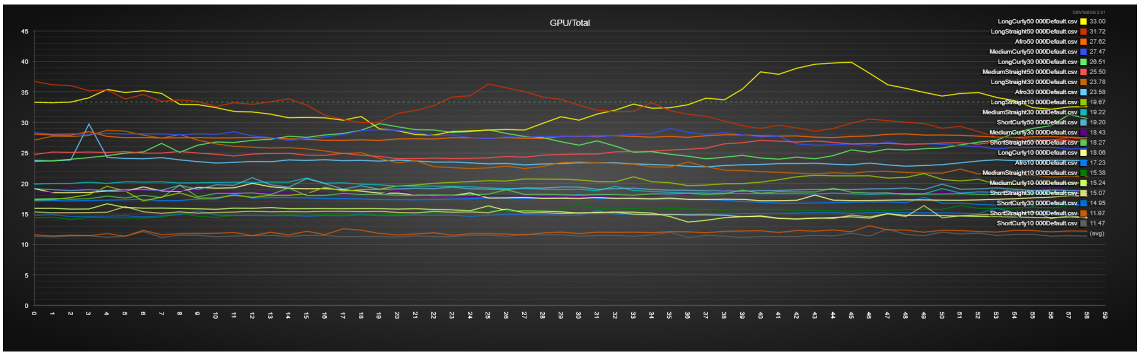

Figure B.2: GPU total time (ms)

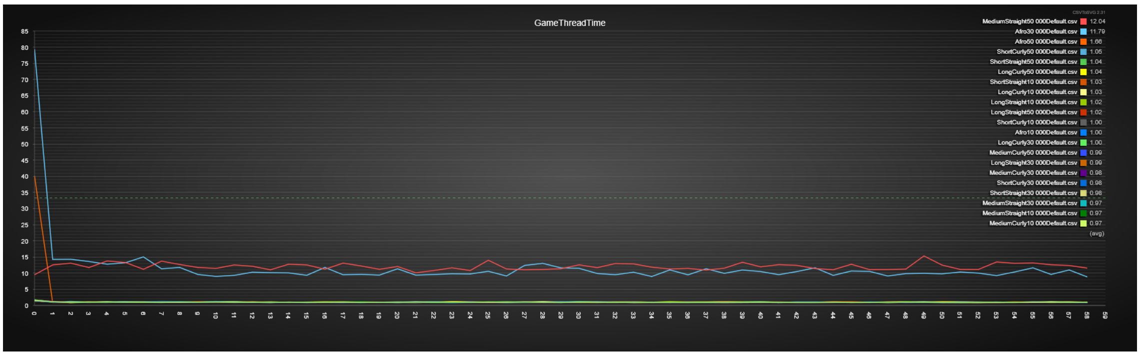

Figure B.3: Game thread time (ms)

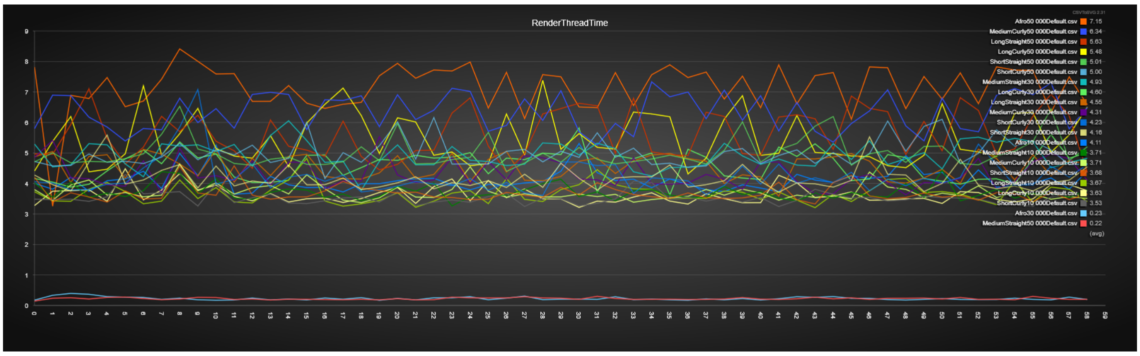

Figure B.4: Render thread time (ms)

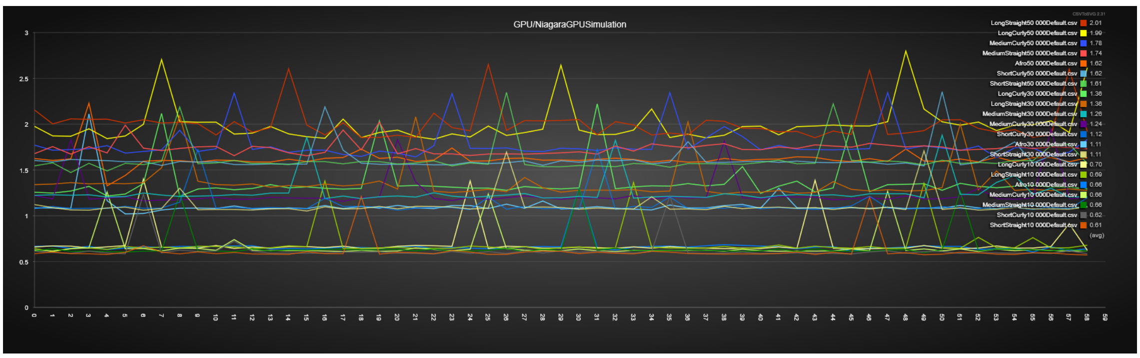

Figure B.5: Niagara GPU simulation time (ms)

Appendix C

Detailed vote distributions

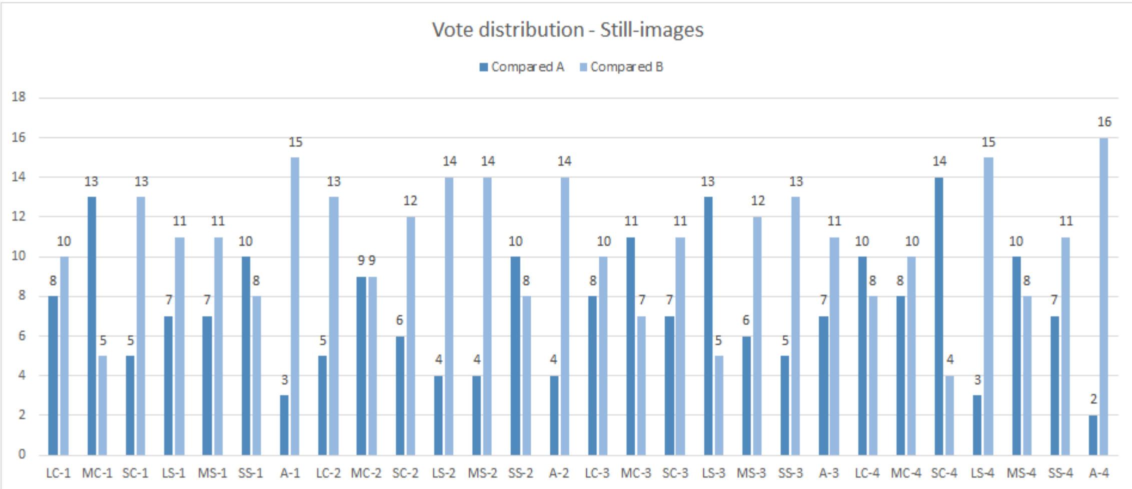

Figure C.1: Chart showing vote distribution on all image-pairs

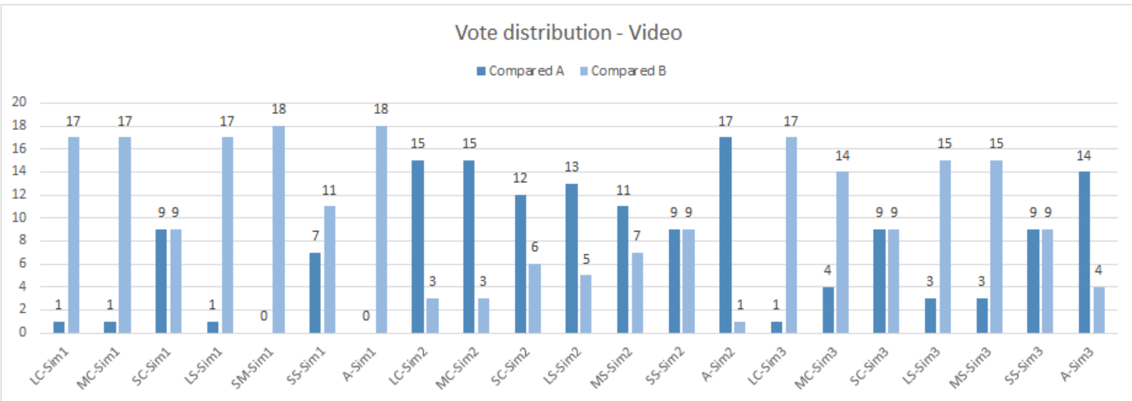

Figure C.2: Chart showing vote distribution on all video-pairs

Appendix D

Invitational letter

Invitation and information for participation in study "Perceived Realism Of Strand-based Hairstyles Simulated"

Recent advancements in technology allows for real-time simulation of hair strands which provides a highly realistic representation of digital hair. In order to determine how different simulation settings applied to different types of hairstyles may impact the perceived level of realism in strand-based hair, you are invited to partake in a study where participants are asked to compare pairs of images and videos. The images and videos depict hairstyles where different simulation settings have been applied and for each pair that is being presented, participants will be asked to choose which they perceive as the most realistic. The study is performed online via Google Forms and participation is estimated to take about 15-20 minutes.

By combining the data collected from the survey with performance tests, the study aims to provide guidelines for how to increase the accuracy of physical simulation for different types of hairstyles without sacrificing computer performance.

Participation in the study is completely voluntarily and participants are free to opt out without giving a motivation at any time during participation. You are required to be at least 18 years of age for participation and have access to a desktop or laptop computer. Participants also need normal or corrected-to-normal vision in order to participate. To assure that the videos included in the study can be watched at the highest resolution without stuttering, a reliably fast internet connection is recommended.

Participation is confidential and the only data collected for the study is the participant's preferences for different simulation settings. This data will be averaged and presented as part of a bachelor thesis done by a student at Blekinge Institute of Technology.

Due to videos being hosted on Youtube.com, some analytics data may be made available to the student performing this study. This data will not be included in the study whatsoever and cannot be used to identify any individual participants. All collected data can only be accessed by the student who performs this study and will be destroyed upon completion of the thesis. Data collection will be performed between 6th of May until 11th of May.

Please follow this link if you are interested in participating:
https://docs.google.com/forms/d/e/1FAIpQLSdkt3l9ZwZSY_J4n05NC6qTm5xGP7Q6njw3KligIMom7bU5iA/viewform?usp=sf_link

Feel free to reach out at any time if you have any questions.

Student performing the study:
Jennifer Larsson-Ståhl
jesh17@student.bth.se

Supervisor:
Veronica Sundstedt
vsu@bth.se

Faculty of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden