Volumetric Intelligence

A Framework for the Creation of Interactive Volumetric Captured Characters

Victor Pardinho
Department of Media
Aalto University

2018

Volumetric Intelligence

A Framework for the Creation of Interactive
Volumetric Captured Characters

Victor Pardinho

Thesis advisor: Pia Tikka
Thesis supervisor: Markku Reunanen

Master's Thesis
New Media Design and Production
Department of Media
School of Arts, Design and Architecture
Aalto University

Helsinki, 2018

Aalto University
School of Arts, Design
and Architecture

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi
Master of Arts thesis abstract

Author Victor Pardinho

Title of thesis Volumetric Intelligence: A Framework for the Creation of Interactive Volumetric Captured Characters

Department Department of Media

Degree programme New Media Design and Production

Year 2019 **Number of pages** 83 **Language** English

Abstract

Virtual simulation of human faces and facial movements has challenged media artists and computer scientists since the first realistic 3D renderings of a human face by Fred Parke in 1972. Today, a range of software and techniques are available for modelling virtual characters and their facial behavior in immersive environments, such as computer games or storyworlds. However, applying these techniques often requires large teams with multidisciplinary expertise, extensive amount of manual labour, as well as financial conditions that are not typically available for individual media artists.

This thesis work demonstrates how an individual artist may create humanlike virtual characters – specifically their facial behaviour – in a relatively fast and automated manner. The method is based on volumetric capturing, or *photogrammetry*, of a set of facial expressions from a real person using a multi-camera setup, and further applying open source and acessible 3D reconstruction and re-topology techniques and software. Furthermore, the study discusses possibilities of utilizing contemporary game engines and applications for building settings that allow real-time interaction between the user and virtual characters.

The thesis documents an innovative framework for the creation of a virtual character captured from a real person, that can be presented and driven in real-time, without the need of a specialized team, high budget or intensive manual labour. This workflow is suitable for research groups, independent teams and individuals seeking for the creation of immersive and real-time experiences and experiments using virtual humanlike characters.

Keywords virtual characters, enactive avatar, volumetric capture, photogrammetry, real-time interaction, real-time engines

Acknowledgments

I would like to thank my advisor research professor Pia Tikka, for the years working together and the great knowledge, guidance and support. My supervisor Markku Reunanen for the immense help to finalise the work, giving valuable feedback and supervision. The Media Department and teaching staff, specially Rasmus Vuori, Nuno Correia and Petri Ruikka for all the inspiring conversations and ideas.

I would also like to thank all the great friends I did during these years at the department, in special Andre Vicentini, Fernando Visockis and Babi Brasileiro for being good friends and making me feel at home. Camilo Carranco, Mathijs Provoost, Otto Montell and Alex van Giersbergen for all the fun times. Also, big thanks to my family and friends back home for always being a great support even at such a distance.

This thesis draws from Pia Tikka's Enactive Cinema (2008) framework and the proof of concept of Enactive Avatar (Tikka, Hjorth et al.2012). The thesis summarizes the research has been made possible with the Aalto Starting Grant for Dr: Pia Tikka research group, during the author's affiliation at the Crucible Studio, Department of Media of Aalto University 2016–2017.

Since 2018 the work has been conducted in collaboration with the Aalto Studio's Virtual Cinema Lab, and Dr. Pia Tikka's Enactive Virtuality research group at the Baltic Film, Media, Arts and Communication School and the Centre of Excellence in Media Innovation and Digital Culture, Tallinn University, funded by EU Mobilitas Pluss Top Researcher Grant by Estonian Research Council (2017-2022).

During 2018 the work has been supported Dr. Pia Tikka & Team funding from the Finnish Cultural Foundation, Huhtamäki Fund. The exploration of the pipeline for commercial purposes has been supported by DigiDemo funding for production company Oblomovies Oy, received by The promotion Center of Audiovisual Culture (AVEK).

Table of contents

1. Introduction

Imagine an immersive virtual reality setting. You will find yourself in the prison cell of some unrecognized country, locked up with a distressed person. You know him and he knows you. By the tone of the voices around you realise the situation is serious. Your feelings and bodily reactions provoked during the encounter are the only way to interact with the fellow virtual being in front of you. His facial expressions and body seems to respond to your feelings. Soon you realise that your emotional states will determine the fate of your fellow prisoner.

1.1 Mission Statement

Since the 1970s, due to an increasing advancements in computer systems and digital image generation, the virtual and tridimensional representation of human faces became a strong subject of research in the fields of computer graphics, media and related practices (Parke, 1972). After almost five decades of research as well as computational, technological advancements, this continues to be one of the greatest challenges, both from the practical and psychological points of views.

This thesis has two fold aim: to cover a range of most relevant previous work conducted on this field, and to describe a path of exploration of technical solutions for creating a responsive humanlike facial behaviour for a 3D CG virtual character based on the capture of a real person. In particular, it concentrates on discussing a specific type of virtual character that is driven by unconscious biofeedback from the user, namely, that of Enactive Avatar, and real-time artificial possibilities to drive such virtual characters using game development technologies and creative coding solutions.

"Volumetrtic Intelligence" is a term that aims for a connection of the technique of volumetric capture to the term "Artificial Intelligence". Since the mission of the workflow demonstrated on this thesis work is in part to create real-time possibilities that will open connections between both volumetric captured characters and Artificial Intelligence systems.

1.2. Actor performances and technical developments

Looking back to the history of capturing human performances, we may conveniently start from the birth of cinema. Actor's performances have been captured since the invention of the camera and screen projections, the main attention turned to human activities and people in domestic or day-to-day environments (Lumière Brothers, 1895) but also to the imaginary human activities, like men flying to the moon (Méliès, 1902). The following transition from a live theatre performance on the theatre stage to a "screen-based" performance required the actors to adapt to new ways of producing the desired outcome, for instance, minimizing their facial expressions in a close-up shot, or being able to repeat their actions exactly the similar manner throughout the several takes in different image sizes, from wide shots to close ups (Doane, 2013). Recently, the awareness of the quality of the actor's performance that is particularly targeted for an animated CG character has given arise to research projects, for instance the Actors and Avatars interdisciplinary project, in ZHdK by prof. Anton Rey (2018).

New demands for capturing actor performances also has brought significant changes to the equipment used in media productions.. Furthermore, accelerated technological inventions and advancements have allowed new artistic practices to emerge.

With the development of 3d software technologies in the field of visual special effects applied in films and cinematic games, it has become a common practice to use in the film productions techniques such as green-screen, motion capture, and digital content interacting with real actors (Monaco & Lindroth, 2000).

The first digitally produced characters date back from 1972 with the works of Fred Parke. The most humanlike virtual character performance as of today, can be considered to be Digital Andy Serkis performance introduced by Epic Game and 3Lateral (2018).

Yet, many of the recent achievements in terms of creating humanlike virtual characters come from big budget cinema and game productions. One of the research motivations in this thesis work is, how can one produce, using current technologies, believable characters within a framework of less manual work and lower costs, when also taking advantage of capturing techniques and digital software automation and algorithms. In the following, I will describe the process that was explored, in detail.

1.3. State-of-art in the field and main psychological and practical challenges

The human face and natural facial expressions are in the core of human social interaction. In order to succeed in creating a computer-generated experience where an artificial face and a human can communicate in a way that resembles or replaces a natural interaction between two humans, a range of challenges related to natural humanlike facial behaviour still have to be resolved.

In 1990, at the dawn of digital computation of human facial dynamics, Lance Williams pointed out the challenges. The human face is an extremely geometric form and possess a complex biomechanics system that can be of high complication to mode and simulate (1990). Today, the challenge is the same. The natural human face is complicated to model as a still head, but it is even more challenging to

animate so that the facial expressions look natural. In a natural face, all facial movements are produced by multiplicity of complex muscular interactions. To artificially produce similar behavioural dynamics requires to produce and animate several skeletal and muscular forms.

> All of these problems are enormously magnified by the fact that we as humans have an uncanny ability to read expressions — an ability that is not merely a learned skill, but part of our deep-rooted instincts. For facial expressions, the slightest deviation from truth is something any person will immediately detect. (Pighin, 2002, p. 1)

However, despite of all the challenges, a need for a tridimensional realistic human face in virtual and immersive environments has persisted.. Techniques to achieve this goal are required in a wide range of applications in high-end productions like movies, animations and computer games. Also at the increasing development of technologies such as Virtual Reality (VR) or Augmented Reality (AR), for entertainment, training and commercial fields in the last decade (Zollhöfer et al., 2018).

Furthermore, when a viewer is placed in a virtual dramatic situation where the unpredicted behaviour of other person, i.e., simulated digital actor, affects the situation in unexpected ways, this adds a new dimension and a deeper layer of meanings to the experience. In the field of games, convincing virtual characters rejuvenate with the ability of interacting in a realistic way within the player (Kalra et al., 1998). In an immersive environments like VR, this feeling becomes a strong factor for the sense of presence.

Accordingly to Zollhöfer et al. (2018, p. 1): “Human faces occupy a very central place in human visual perception since they play a key role in conveying identity, message, emotion, and intent”.

In this thesis work, will be proposed a workflow for the creation of a realistic virtual character utilizing volumetric capture techniques and the use of accessible hardware and software. Although the creation of realistic digital humans is very challenging using traditional 3D modelling techniques, with the advances in volumetric capture techniques, such as photogrammetry and volumetric video, we can propose a new approach for this goal.

Volumetric scanning techniques provide a fast method for constructing tridimensional meshes, utilizing a live-action capture approach opposite to computer modelling tools. Contemporary software that utilizes automation and artificial intelligent based algorithms can to a great extent automate most of the hand work needed for the creation of a digital character.

In this work, I will demonstrate and walk the reader through a process utilizing capturing concepts and techniques in order to create a realistic virtual character from a real person. I will focus on automated systems instead of manual work and specialized labour. My work will demonstrate how, with the tools and the development of volumetric capture, we can manage to create a real-time animated realistic character in a faster and easier manner, by minimizing the manual animation work and significantly speeding up the process. We will also introduce the use of such a virtual character into real-time environment, in which machine learning (Artificial Intelligence), facial motion capture and other recently developed methods can be applied to production of such a character, supporting character’s unpredictable automated, read, more humanlike behaviour.

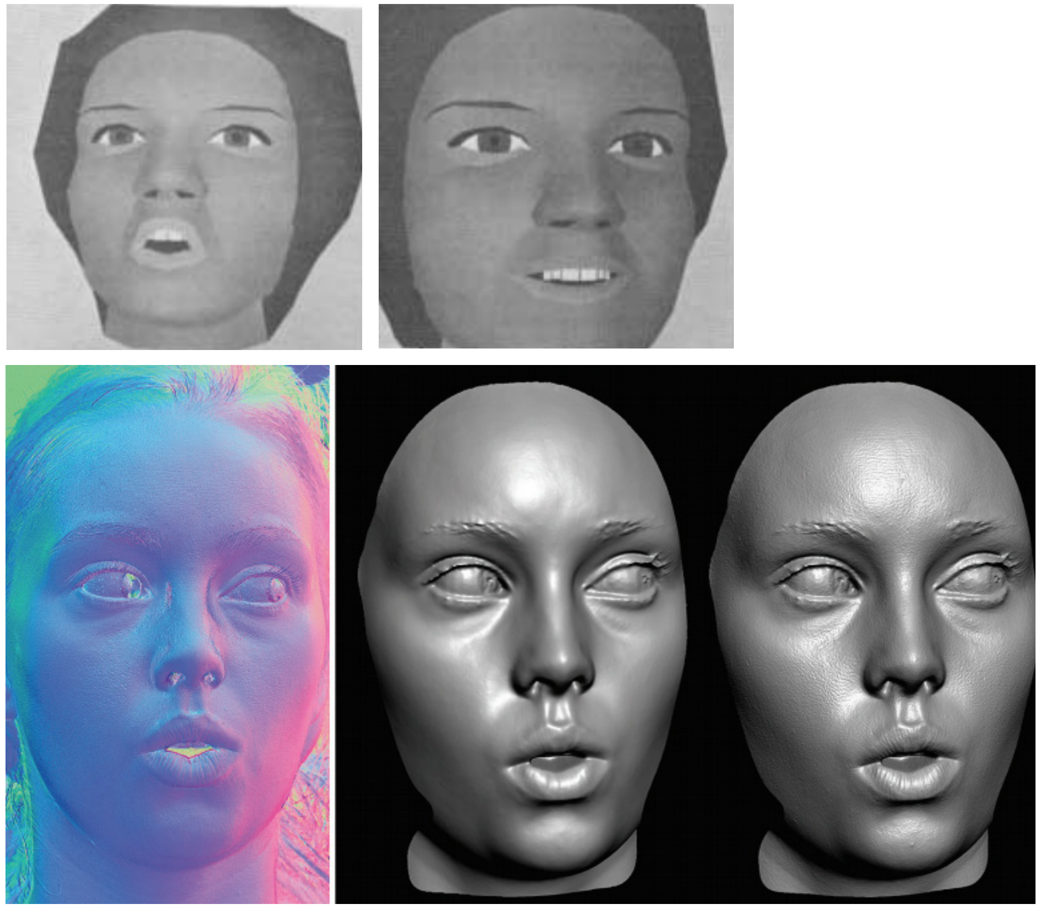


Figure 1: Frederick Parke (1972) and Digital Emily (Alexander, 2010).

1.4. Related Works

A large body of studies have been accumulated in the field of realistic human modelling and animation over the years. However, making use of a captured workflow based on automation and algorithms instead of traditional tridimensional modelling pipelines, is a relatively recent practice. This work is highly influenced by the achievements of the USC Institute for Creative Technologies Especially Digital Emily (Alexander, 2010) and Digital Ira (von der Pahlen et al., 2014) can be mentioned. as works, where high-end hardware and software were created in order to achieve a realistic virtual human being,. In our case, we seek for the same outcome but utilizing low-end and off-the-shelf solutions available for individual media artists.

The work of Zollhöfer et al. (2018) also shares similar aspects with our workflow. In their article "State of the Art on Monocular 3D Face Reconstruction, Tracking, and Applications" they evaluate different algorithms for the capture and reconstruction of human faces, introducing also possibilites of cpaturing actors' movements. With a strict engineering focus, their work communicates very well with the automated approach proposed in this thesis.

Out work relates to the extended work of Mark Sagar et al. (2016) at the Laboratory for Animate Technologies of Auckland University , especially their work conducted on the ways the viewer may interact with virtual characters. The same is true also with facial game technologies, especially volumetric technologies used in games like L.A. Noire (Star, 2011) and the achievements of Ninja Theory's Hellblade (2017). From the scientific view-point , the research by Mark Cavazza and others (2002) shares similarities with our work, particularly related to character based interactive storytelling.

This work also seeks to relate the researches and developments in sequenced volumetric capture solutions, especially the experiments from the volumetric capture community, such as the work of Or and Anlen (2018), Dou et al. (2017), Scatter (2017), as well as volumetric capture studios as Microsoft Mixed Reality Capture Studio (2018) and Fraunhofer Institute's Volucap Studio (2018).

The uncanny valley discussions, specially the psychological research by Jari Kätsyri et al. (2015), Aline W. de Borst and Beatrice de Gelder (2015) and Rachel McDonnell et al. (2012) are also of great value to this work, bringing and opening discussions on the way humans perceive and interact with virtual characters. This discussions proved to be of high complexity over the years and is out of the scope of this thesis, but still fundamental for any production focusing on virtual characters with a realistic approach.

1.5. Enactive Cinema and the concept of Avatar

This thesis work is part of Pia Tikka's Enactive Avatar project, a follow-up of her Enactive Cinema (2008) framework.

"Enactive cinema takes the simulation one step further by letting the viewer's experience influence the narrative in real time. Being enactive refers to engagement that is more holistic than being interactive." (Tikka et al., 2012, p. 2)

It also follows-up the work of Tikka and Hjorth et al. (2015), using 4K stereo footage for a proof of concept for video captured image, integrating interactive eyes as well as adaptive background (Fig. 2). This approach has been later further researched and developed in my thesis.

Following Tikka, I will systematically use the concept of avatar to refer to a virtual character that has been created by photogrammetry capturing process of a real actress of actor. Typically, especially in games, avatar concept refers to the agency or representative identity of the player or user in the virtual story world. Thus, my concept of avatar in this thesis (following Pia Tikka p.c.) is regarded as a synonym to the notion of humanlike virtual character, and should be differentiated from the more conventional use of the concept.

In contrast to the general concept of human-computer interaction, the concept of enaction conceives in a continuous, ubiquitous and even unconscious interaction between the user and the virtual subject. An enactive virtual character also shares cognitive feedback, living and acting with the system instead of just responding to it (Kaipainen et al., 2011).

Therefore, the enactive interaction unfolds no to by direct ways such as clicking interface elements or devices, but it emphasizes unconscious interaction, being those based on psychophysiological data or interactions such as eye-gazing and subtle movements (Tikka et al., 2006). In this work we also open the possibility for an artificial intelligent system, using machine learning techniques to enhance the enactive relation between the avatar and the viewer.

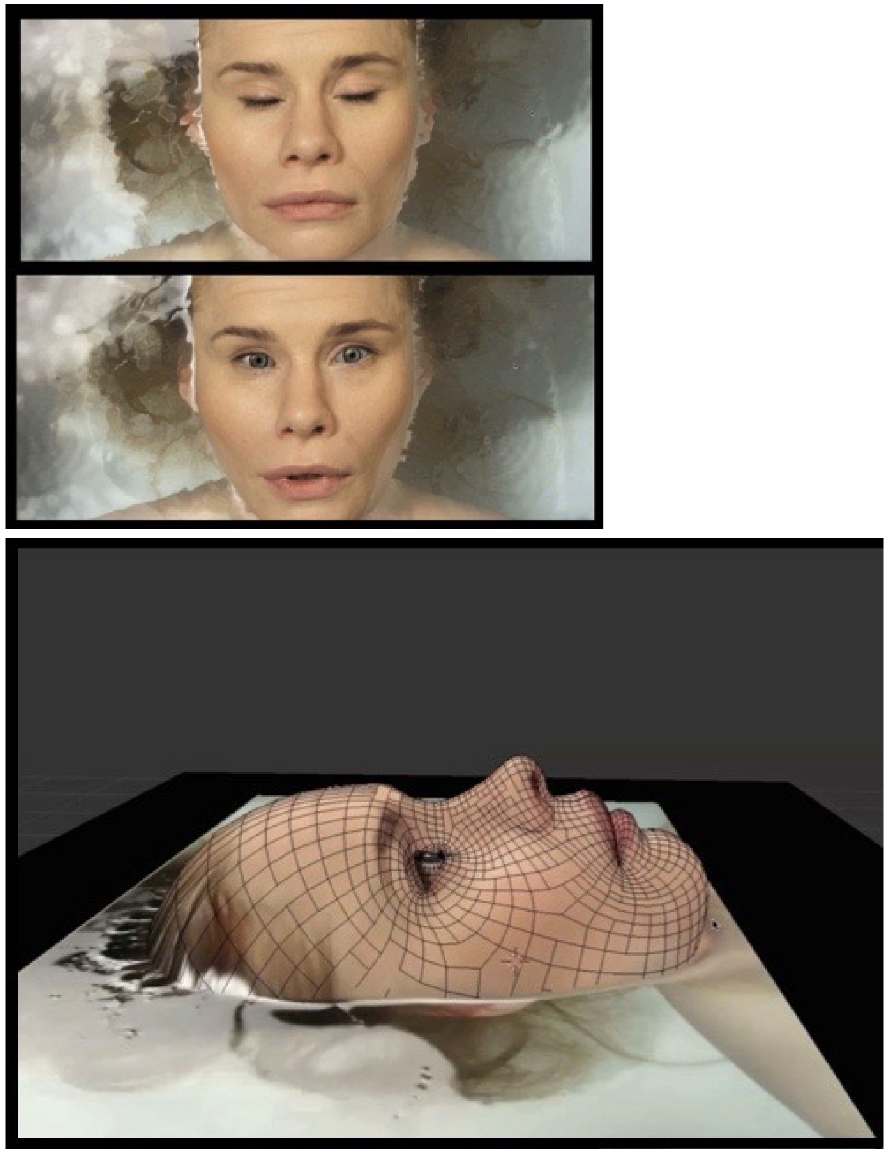


Figure 2: Enactive Avatar first prototype by Tikka & Peter Hjorth (2015).

2. Capturing process of real actors for the production of tridimensional virtual characters

The journey for the creation of a virtual character based on a real actor's data starts with the capturing phase. This phase is a crucial part for the whole workflow, since good captured data is essential in order to build good results on succeeding steps. It is important to keep in mind that in order to correct a bad captured data, most of the times, the only solution is to re-capture it, which will cause delays and extra expenses for the production.

The capture process described in this thesis work consists of the capture of a real person in a tridimensional space by using a technique called photogrammetry which "encompasses methods of image measurement and interpretation in order to derive the shape and location of an object from one or more photographs of that object" (Luhmann et al. 2007, p 2). Meaning that, by taking several photographs of an object from different angles, it's possible to find the measurements between the position of the camera and the object, therefore, creating a tridimensional shape.

The content generated with photogrammetry techniques allow multiple viewing points to the virtual character, when located in virtual and immersive tridimensional environments (e.g.Virtual Reality, or Augmented Reality). By using photgrammetry, the participant is offered has a possibility to walk around the virtual character, to come closer, and even to interact with the character in a relatively natural manner.

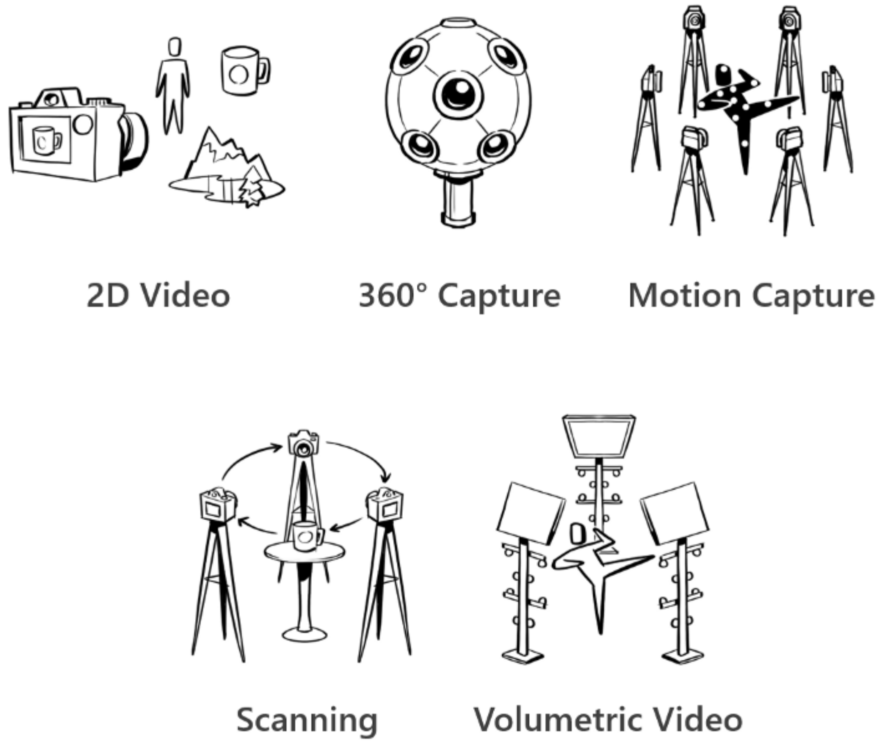


Figure 3: Different ways of capturing an image for an immersive environment (Vitazko, 2017).

2.1. Photogrammetry, techniques and equipments

For the capture of a real space or person in a volumetric manner the use of photogrammetry can be applied.

This technique is mainly used for measurement mathematics, but with the advance of photography cameras and possibility to take high-quality images that later can be projected to the tridimensional shape. In the latest years this practice has been appearing as an aesthetical choice besides its original measurement solution in several tridimensional entertainment fields, like movies and games (Statham 2018).

In order to create a tridimensional object using photogrammetry, the first equipment that is needed is a high quality photo or video camera. Traditional DSLR cameras are the most widely used and can provide great results (Ruan et al., 2018).

Figure 4: Photogrammetry capturing process of forest elements for a virtual game (Poznanski, 2014).

For the capture of spaces and environments, this technique is done by first taking several pictures of the environment requested at the most variable angles possible and at the same period of time, in order to avoid light changes. For the capture of a person, it is also possible to capture the subject using only one camera and taking several pictures around the person as fast as possible, but usually it doesn't provide enough quality at the post-capturing phase.

In order for a photogrammetry capture to generate good data, the subject must not be moving, which is perfect for static objects or physical places. However, in the case of a person, even the movement of breathing is already enough to not provide a good quality tridimensional model, especially when the subject needs to perform strong facial expressions and hold her or his facial expression while the capture occurs.

For this reason, the photogrammetry process of a human being becomes exponentially more complicated than the capture of a static object, since even a breathing movement is enough to create serious data artifacts. The solution must commonly used is know as a photogrammetry setup, also photogrammetry booth or studio (Fyffe et al., 2015; Fig. 5), which consist of a set of cameras built around the subject in different heights, powered with a system that takes photos in each camera at the exactly same time. Therefore, a data-set of pictures from the person in every angle at the exactly same point in time is given. And without movement feedback, the photogrammetry algorithm can work perfectly and generate a high quality tridimensional model without much errors or artifacts later in the process.

In this process, also the lightning of the subject is captured, for this reason, it is necessary to use white uniform light, like a traditional studio diffuse light, in order to avoid difference of lightning conditions from the moment of the capture to the virtual experience later on. By capturing with a white diffused and uniform light, it becomes possible to relight the tridimensional object and apply post-processing effects to achieve any choice of light colour and effects at the virtual 3D environment.

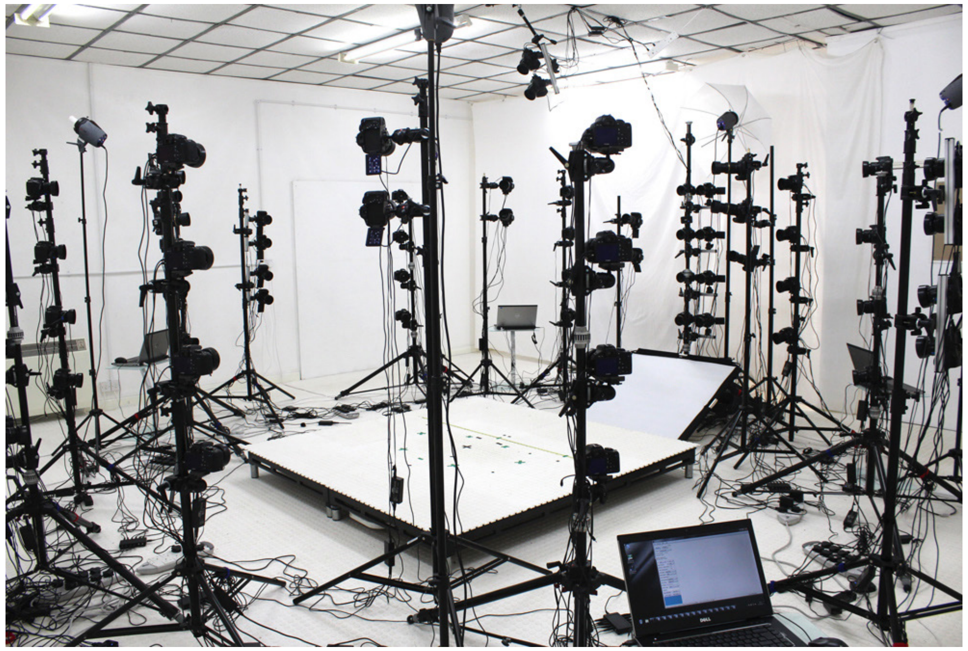


Figure 5: An example of a Photogrammetry Setup containing 72 cameras (Tilbury, 2012).

2.2. Character capture workflow and best practices

The workflow for capturing an actor or person in a photogrammetry booth for the creation of a virtual character can be quite different from capturing in a traditional film or video. In this case, the capture of different key facial expressions to be generated in 3D space was the focus, and not a moving performance. The goal is to create a library of key expressions to be later generated and manipulated in tridimensional space.

The choice of expressions to be captured can vary from project to project, therefore it is important to keep in mind which kind of facial expression is needed for a particular project. For example, if later in the project the team decides that they need the virtual character to have a sad expression with eyes closed, but this was not captured, it can be tricky and heavily work to create such an expression by 3D modelling and direct manipulation of the virtual subject, which will not deliver such good results as if it was captured from the real person.

Therefore, if the team is aiming for a scripted linear experience in which the facial expressions are already set, then it should be aimed at making a good capture of that data-set scripted. However, if the aim is to create an emergent virtual character with several different expressions, then it can be advised to capture as much expressions as needed. It is also possible to follow a Facial Action Coding System (FACS) (Ekman, 1997) in order to know which exactly expressions to be captured, especially if the team plans to use a motion-capture device later in the virtual environment.

One common set of facial expression being used lately in real-time environments are Apple's iPhone X facial blendshape data-set (Apple, 2017), which consist of fifth one different facial movements that can be later be driven through Apple's facial motion capture solution embedded on their iPhone X camera hardware (Zollhöfer, 2018).

In the case of the projects mentioned in this thesis, the facial expressions were acquired using the Triplegangers photogrammetry capture data-base (Triplegangers, 2017; Fig. 6), based on a set of expressions chosen to be used for a proof of concept of a realistic emergent virtual character or avatar.

The expressions chosen were twelve: Neutral, Neutral Eyes-Closed, Face Compressed, Snarl, Look Up, Chew-Right, Phoneme-Hard, Happy, Surprise, Pain, Disgust and Rage. Based on Ekman (1997) guides and the project script.

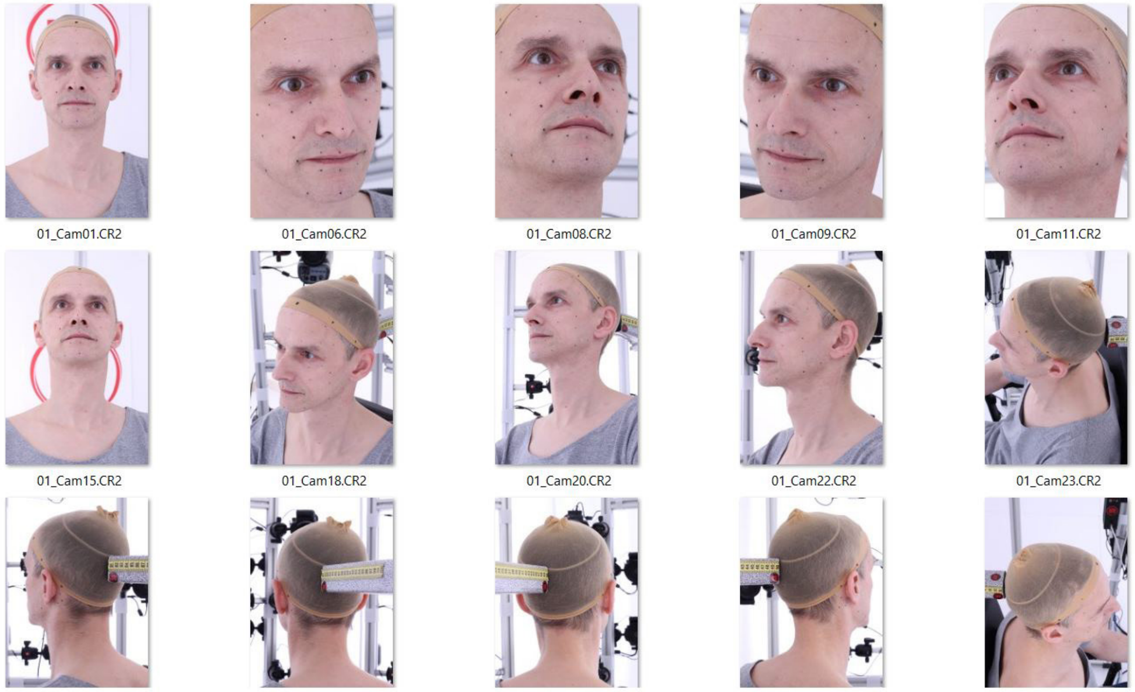


Figure 6: Data-set photographs from a Neutral facial expression taken in a Photogrammetry setup.

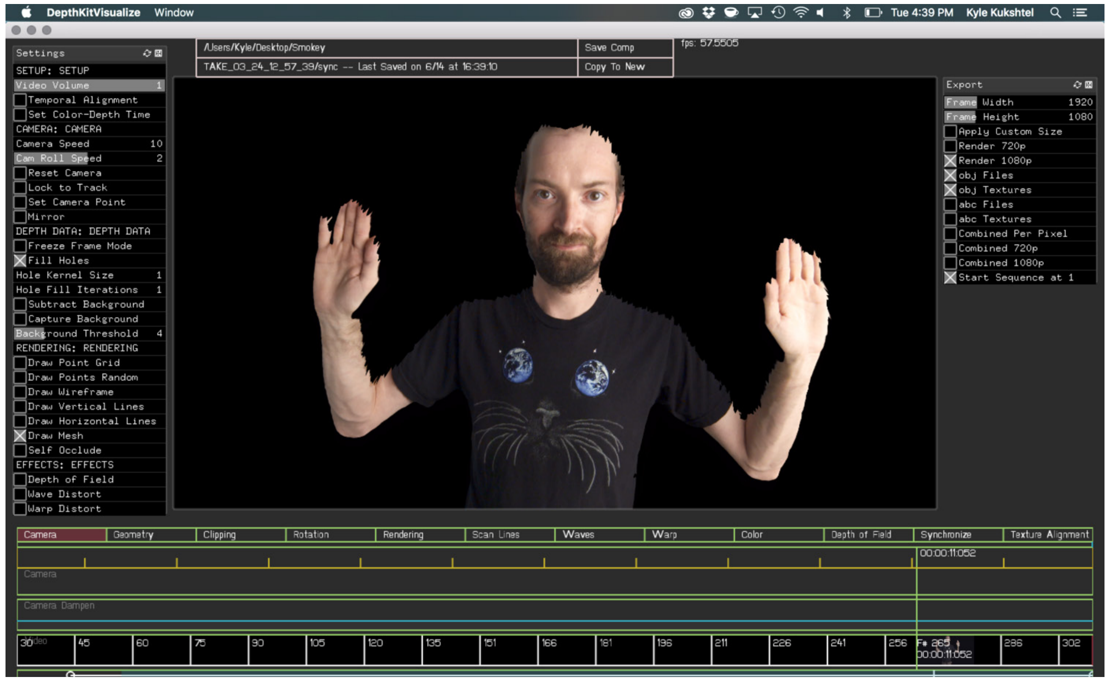
2.2.1. Structured Light and other techniques

With the advances in photogrammetry capture for entertainment experiences, other techniques and possibilities started to be developed by filmmakers and audiovisual technology researchers, and produced in order to avoid the need of a photogrammetry booth or a studio solution. One famous technique is known as Structure Light:

> A sequence of known patterns is sequentially projected onto an object, which gets deformed by geometric shape of the object. The object is then observed from a camera from a different direction. By analyzing the distortion of the observe pattern, i.e. the disparity from the original projected pattern, depth information can be extracted (Sarbolandi et. al, 2015, p. 5).

Used in depth-cameras hardware like Kinect (Microsoft 2010) and the Intel Realsense (Zollhöfer, 2018), this is a great solution for prototyping and fast-paced testing of a volumetric capture experiment. However, for production-ready virtual characters, it is not advised since it collects a good amount of noise and unclean data, which need to be cleaned and heavily post-processed to be able to be used in production. This technique is becoming widely used for real-time volumetric capture, also called "holograms" or "volumetric video" in the commercial field (Fig. 7).

Another technique used by independent artists or small studios that can't afford a capturing session at a photogrammetry booth or the time to create a high quality capture by hand, is to use video instead of photographs. By recording a video using a camera dolly or a structure that can rotate around the subject while taking a high-quality and stabilized video can provide a good volumetric capture without the need of a multi-camera system.



/Users/Kyle/Desktop/Smokey
TAKE_03_24_12_57_39/sync -- Last Saved on 6/14 at 16:39:10

Settings
SETUP: SETUP
Video Volume 1
□ Temporal Alignment
□ Set Color-Depth Time
CAMERA: CAMERA
Camera Speed 10
Cam Roll Speed 2
□ Reset Camera
□ Lock to Track
□ Set Camera Point
□ Mirror
DEPTH DATA: DEPTH DATA
□ Freeze Frame Mode
☒ Fill Holes
Hole Kernel Size 1
Hole Fill Iterations 1
□ Subtract Background
□ Capture Background
Background Threshold 4
RENDERING: RENDERING
□ Draw Point Grid
□ Draw Points Random
□ Draw Wireframe
□ Draw Vertical Lines
□ Draw Horizontal Lines
☒ Draw Mesh
□ Self Occlude
EFFECTS: EFFECTS
□ Depth of Field
□ Wave Distort
□ Warp Distort

Export
Frame Width 1920
Frame Height 1080
□ Apply Custom Size
□ Render 720p
☒ Render 1080p
☒ obj Files
☒ obj Textures
□ abc Files
□ abc Textures
□ Combined Per Pixel
□ Combined 720p
□ Combined 1080p
☒ Start Sequence at 1

Figure 7: Volumetric Video capture software using a depth sensor camera (Scatter, 2017).

This process is still not perfect since, even though it can be captured faster than a hand-handling, there's still a great quantity of movement while the person hold her or his facial expression, but opposite from a hand-held capture, the amount of errors and artifacts in the post-processing phase can be possible to handle without losing much time and resources.

Industrial 3D scanners that use the LiDAR technique, it's similar to the Structure Light technology explained above but in this case the use of lasers are more common, while recording a video or taking photographs in a rotational base (Boehler & Marbs, 2002). While widely used for environments and spaces in the industry field, they are not used or advised to use for human facial scanning since their laser technology can't achieve a proper resolution required for a human face scan, therefore they won't be mentioned further in this thesis.

In the case of using a video capture technique, it is needed to extract the frames from the video footage, therefore the frames of the video are used as if they were photographs taken around the subject. Which is what is needed for the next phase of this workflow.

Light Field technology is another promising possibility to capture live-action performance and transport ot a virtual immersive environment. Strongly researched by the company Lytro and aquire by Google in early 2018, this technique uses an array of cameras that not only sense light intensity but also the direction and bouncing of several individuals ray of lights traveling the space. In this way the algorithi can reconstruct the environment in a virtual space using real footage and light behaviours (Tricart, 2017).

This technique is still in strong prototyping phase since it is of greater technical challenge, but several achievements have been aquired with strong funding from enterprises and research groups in order to miniaturize prototype systems and create possible commercial solutions for distribution and content creation.

2.3. Post-production of the captured data

After capturing the desired real person performing key facial expression using a photogrammetry setup to generate photographs of every angle at the exactly same time, the data set is ready for the post-production of this captured data. The captured actor in this thesis is named "Alan Brooks", it was casted by the leader of one of the Enactive Virtuality project by Pia Tikka, and will be described later in this work.

At this stage, photogrammetry software were used to generate 3D models from the 2D images using a series of processes powered with strong computer vision algorithms. For the purpose of this thesis, I used AgiSoft PhotoScan as the software of choice. AgiSoft was chosen for its simplicity of use and my previous experience in utilizing the tool for my past projects.

In the next sections, I will e describe the processes needed to generate from the photographic data-set into a workable tridimensional virtual character that can be imported into a real-time engine. The tutorial form facilliates understanding of how the process flows and can be re-created by future teams and individuals that seeks the creation of a virtual character based on capturing techniques, opposite to traditional hand 3d modelling based on reference photos.

The following workflow presented contains five main steps: Camera Alignment, Dense Point Cloud, 3D Mesh Generation, 3D Mesh Decimation and 3D Mesh Texturing.

2.3.1. Camera Alignment

The first step for this post-captured data processing workflow is the alignment of the camera position for each photograph captured. In this process, the software searches for common features in each photograph and matches them pixel by pixel by using image analyzing (Agisoft, 2014). Also, by triangulation algorithms, the software will generate the probable position and orientation of the camera at the moment it took the picture (Warne, 2015).

The result is the generation of a sparse point cloud, a collection of 3D points in space and the position and orientation of all the photographs (Fig. 8). This sparse point cloud, as the name indicates, is a sparse reference of the 3D points in space, therefore the data isn't usually exported in this workflow, even though this it is widely used when doing photogrammetry of environments and physical spaces.

In the case of this particular project, I used 22 photographs and this process took around 5 minutes to be concluded, generating a sparse point cloud of 15,988 points.

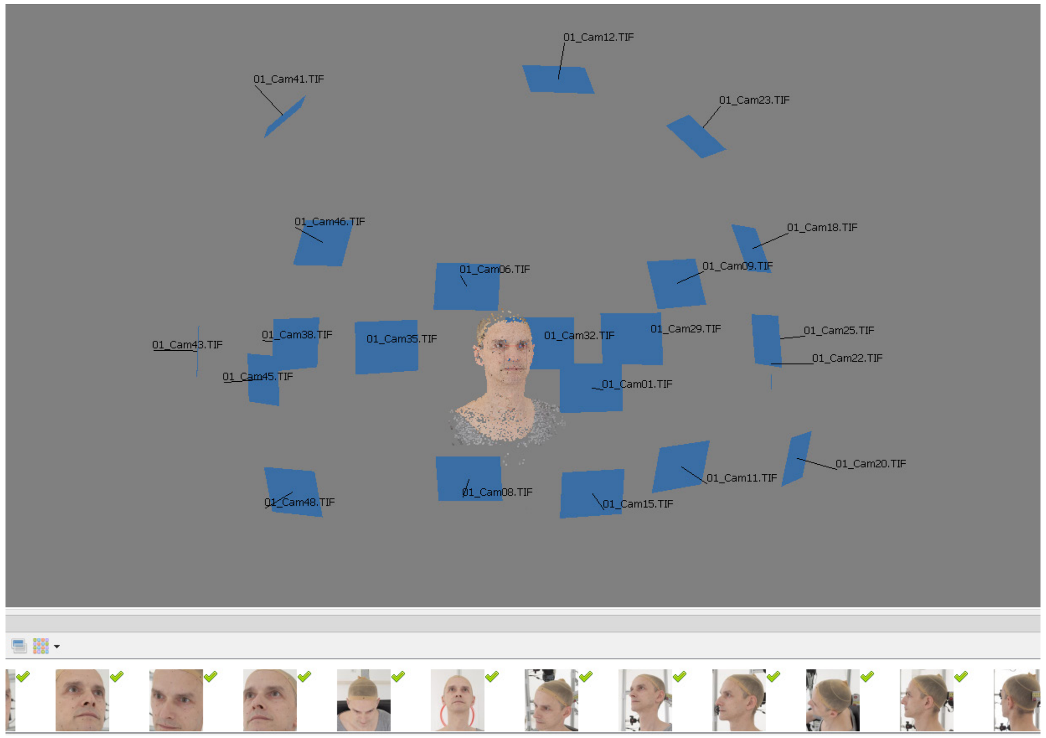


Figure 8: Camera alignment displaying virtually the camera positions at the moment of capture.

2.3.2. Dense Point Cloud

After the algorithm generated a sparse point cloud of our subject based on the photographs taken, it can generate what is called a "dense point cloud". At this process, the application generates depth maps of each photo and together with the information from the last step, knowing the photographs position and the sparse point cloud data, it produces a much higher amount of small 3D points in the space with RGB values created by analyzing the colour data of the photos (Koutsoudis et al., 2014).

Therefore, a dense point cloud, meaning a dense collection of 3D points in the space that forms our subject with colour information, which is crucial for the next steps.

The Dense Cloud generated 13 million points in High quality, taking around 30 minutes to be concluded (Fig. 9). This was the most time consuming process of the workflow.

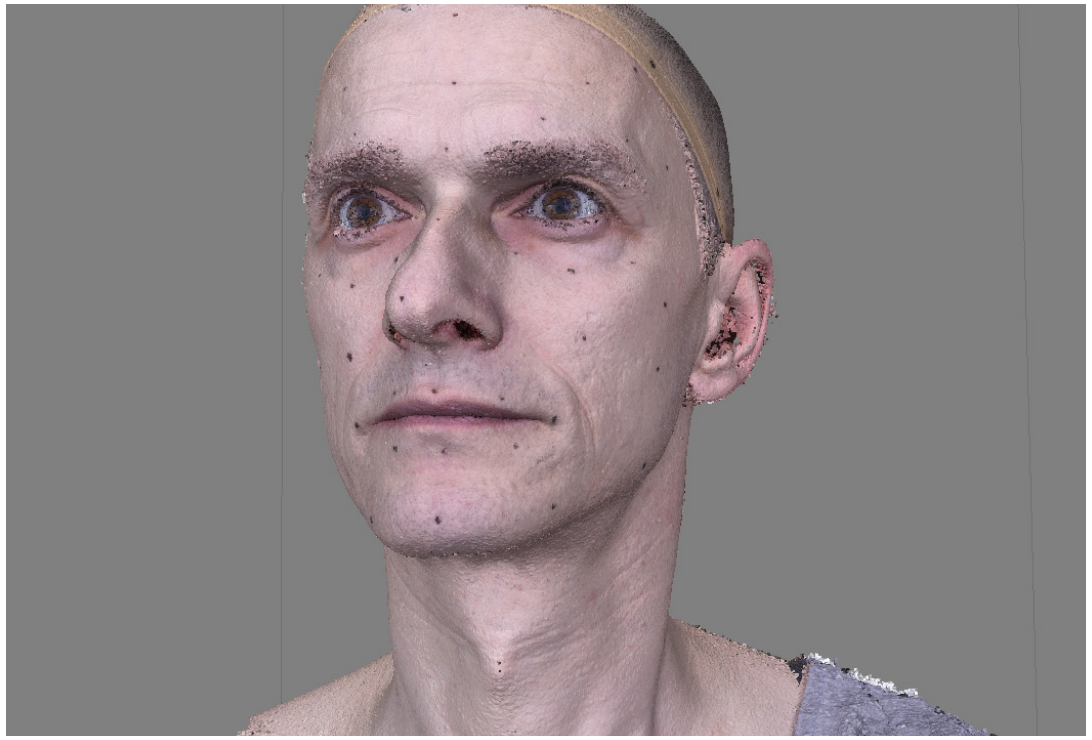


Figure 9: Dense Point cloud of t the scanned head.

2.3.3. 3D Mesh Generation

With the dense point cloud generated containing also RGB values for each 3D point in space, a 3D mesh of our captured subject can be generated. At this point the software provides two options of processes: Height Field or Arbitrary.

The Height Field process is optimized for planar surfaces, usually used for aerial photography processing, for example, using drones to capture the photographs needed for this process. While the Arbitrary process is optimized for closer objects, such as museum artifacts, buildings, and as in the case of this work, the head and facial expression of a scanned person.

At this stage, the software implements a Structure-From-Motion (SfM) and dense multi-view stereo-matching algorithms (Koutsoudis et al., 2014) to create a mesh triangulation and be able to generate a 3D mesh, or 3D model, of the scanned subject (Fig. 10). This process took around 15 minutes to be concluded.

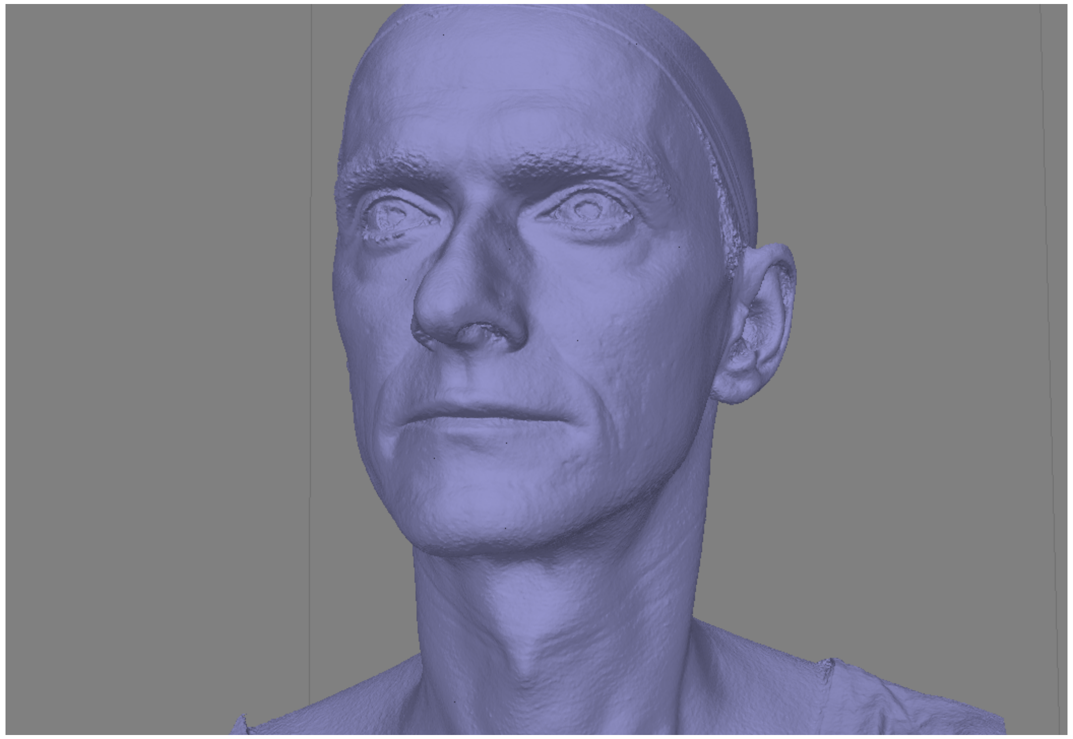


Figure 10: 3D Mesh created using data from the Dense Point Cloud and cameras.

2.3.4. 3D Mesh Decimation

By using the photographs captured at the photogrammetry booth and generating the 3D mesh of our subject, the post-processing of captured data is almost finished. However, there's a crucial step that needs to be made before exporting the new 3D asset, this process is known as Mesh Decimation (Kobbelt et al., 1998).

When the 3D mesh is generated, it is a high poly version of the scanned subject, meaning that the amount of polygons are to hard to be easily workable. In the case of a human head the polygon count of the 3D mesh can get to seven million polygons or more, which makes it highly difficult to work with inside traditional 3D editing software or real-time 3D engines.

Therefore, there's a need to convert our high polygon version into a low polygon, which will be lighter and of easier manipulation on the following steps. This is a usual practice in the field of computer graphics, and the decimation algorithms can be so powerful that after the 3D mesh being textured it is difficult to actually realise the difference between the high poly version and the low poly.

The amount of polygons to be decimated will depend on the particular project, in this case, a decimated the high poly version, which is around seven million polygons into a lower version, which is around two hundred thousand polygons was generated (Fig. 11). It is a big difference in polygon count but visually the 3D mesh doesn't suffer much damage. This process took around 30 seconds to be concluded.

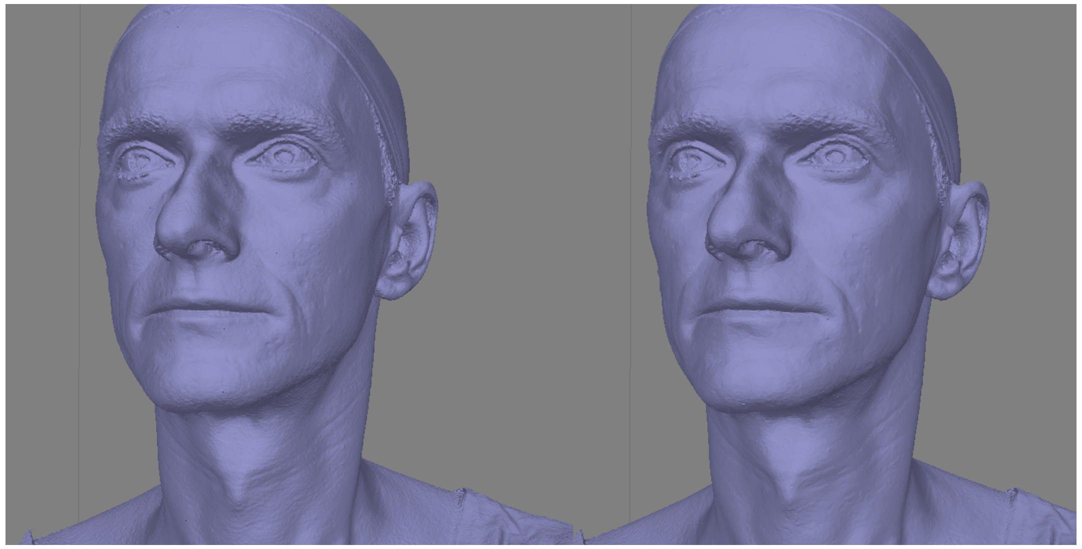


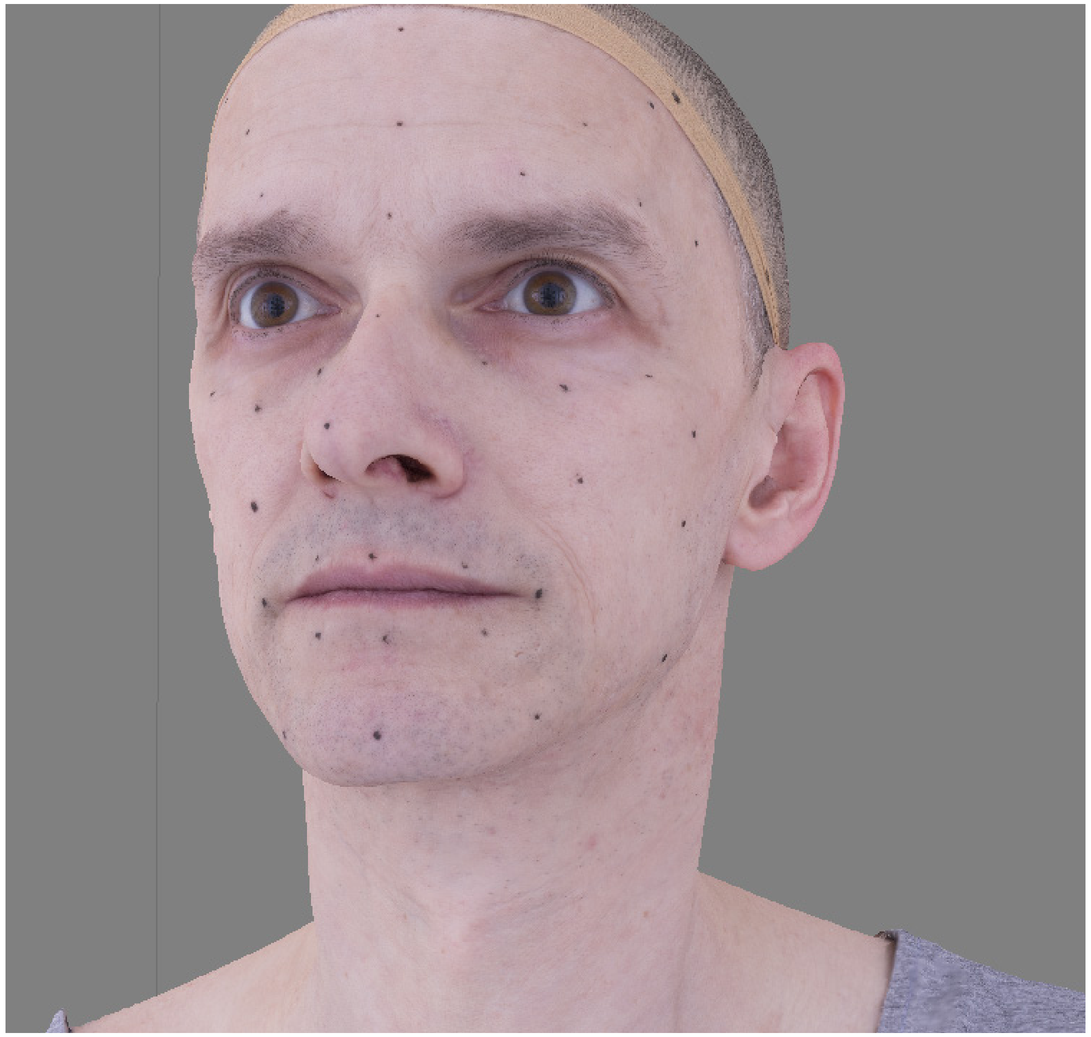
Figure 11: High poly 3D mesh and Decimated mesh side-by-side.

2.3.5. 3D Mesh Texturing

At this stage, a 3D mesh ready to be used from our scanned actress, actor or regular person its given, and with it the possibility to texture the 3D mesh in order to give it a realistic look. This process is also generated using the photogrammetry software, in this case Agisoft Photoscan. At this point, the software has all the data it needs to generate a texture for our 3D mesh: the position of the cameras, the 3D mesh generated from the dense point cloud and the pixel colour value from all the photographs taken.

Therefore, the generation of textures to be applied to the newly created model happens with a set of methods and algorithms. First the software makes a fast colour calibration between the images in order to avoid lightning differences from one photo to another (Agisoft, 2014), it is important to note here that this colour calibration process isn't powerful enough to avoid extensive differences, it is still important to capture the subject in uniform diffuse white light to avoid lightning issues at later processes.

Next, the application will blend the images into one using image blending and morphing techniques and will project the generated image on top of the 3D mesh accordingly to its coordinates in space. This process can be done in several different options, for example a "Generic" option, or mapping mode, which parametrize the texture for arbitrary geometry. "Adaptive Orthophoto" which separates the data into flat surfaces and vertical surfaces creating a texture accordingly. Or "Spherical", as the name indicates, it is optimized for spherical or ball-shaped objects (Agisoft, 2014). In the case of our workflow, the Generic mode was used and the process took around 2 minutes to be concluded (Fig. 12).



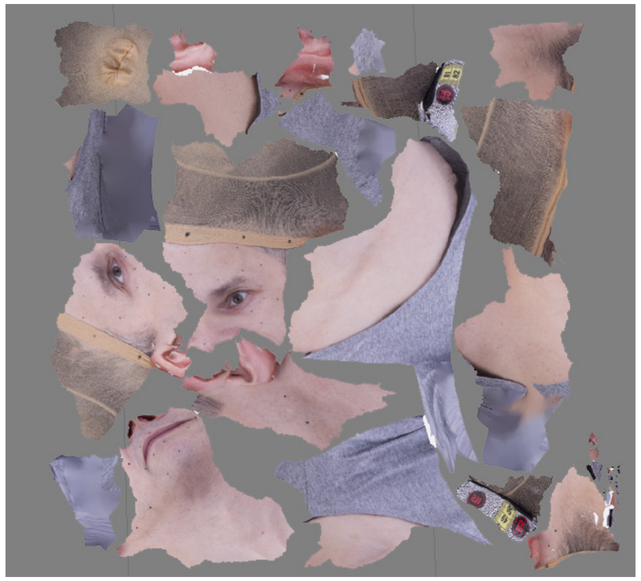


Figure 12: 3D Model textured using a Generic Mapping mode based on the photographic data and the texture source map.

2.4. Data capture post-production conclusions

Within these steps, a 3D asset ready to be manipulated is generated from photographs of a real person. This process can seem to take time and resources but new advances in photogrammetry are proving that it is a much quicker process for the creation of a 3D realistic asset of a human head than modelling by hand. It also avoids the need of a specific expert for the work, as it is common on traditional 3D content creation, which can be time consuming and highly costly (Warne, 2015).

Four the purpose of this work, the process was created for each expression that was captured, therefore in the end of this phase, 3D ready assets of all the facial expressions from our subject are gathered (Fig. 13). It may seem complicated to re-make the process for each expression, but the software of our case has the option of Batch Process (Agisoft, 2014). This means that after doing the process one time, the software can be programmed to redo the process automatically, for each expression, saving great amount of time and resources. After the processes are finished, 3D assets from all expressions are ready to be implemented in real-time, providing a workable and morphable version of our virtual character.

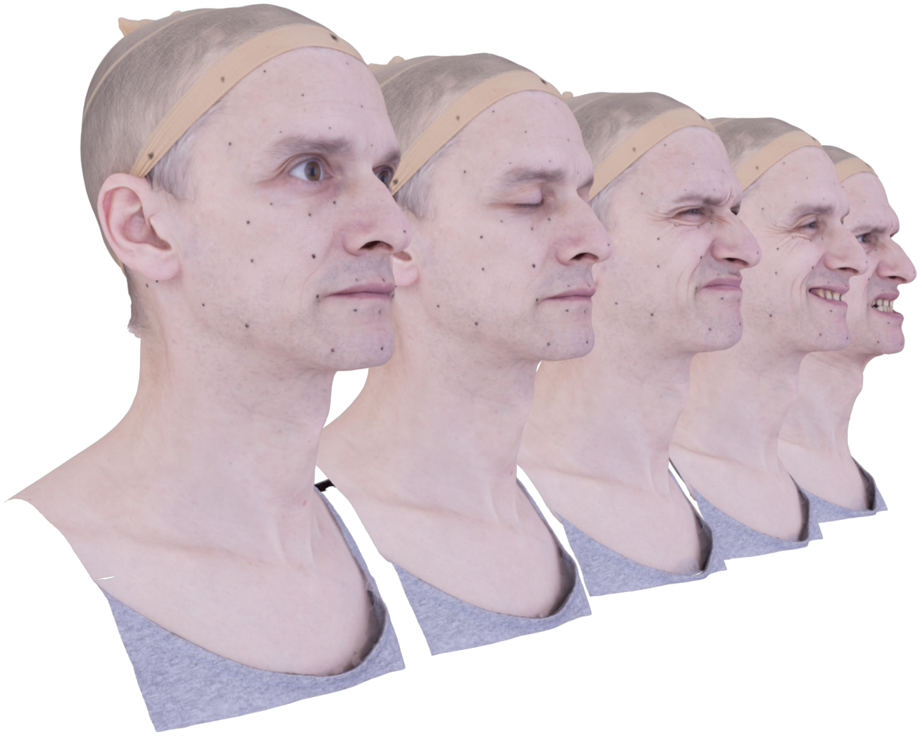
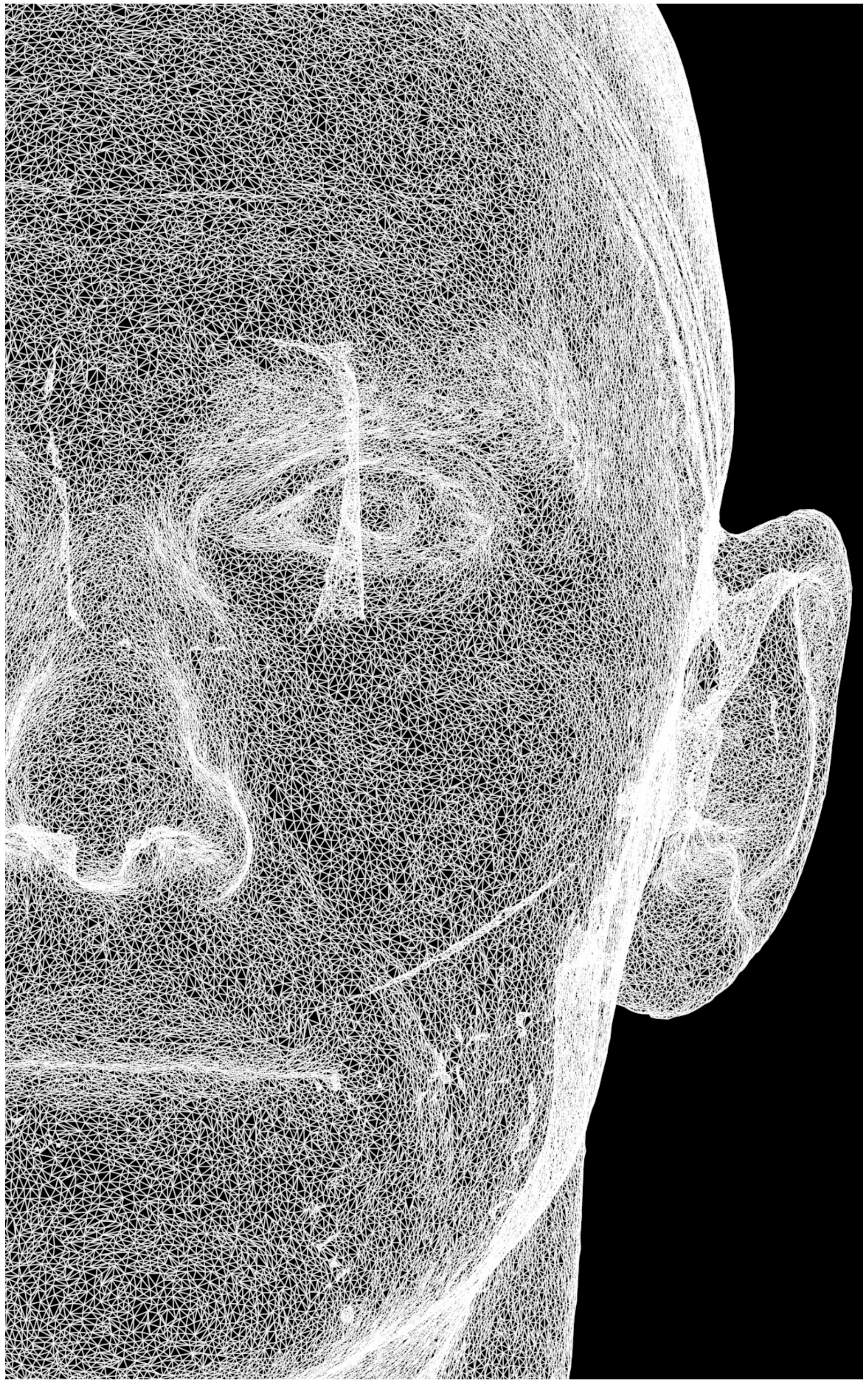


Figure 13: 3D ready assets from different scanned facial expressions.



3. Bringing a captured 3D ready assets inside a real-time virtual environment

With a set of 3D models with different expressions captured using photogrammetry, it is possible to shift the work totally inside a virtual tridimensional environment and apply techniques in order to create a workable character that can run inside a real-time engine and, therefore, being able to interact with the user dynamically.

In this chapter of this thesis work, will be demonstrate how to prepare a virtual character generated from photogrammetry for such an environment, that cointains the same 3D topology information being able to morph between different expressions. This method opens the possibility to create interactive virtual experiences that feels realistic and natural for the user to interact with the virtual character. There's also the possibility to implement real-time techniques for emergent characters using user-data analysis, machine learning and other techniques that will be introduced later in this work.

For the virtual character to be ready for real-time environments, it needed to pass through a workflow that will be described next. This process uses several different software and opposite to the "traditional" handmade process which can take long time and resources, in this thesis, the focus will be on an experimental and automated workflow.

The content generated with these systems are specially created to be reproduced in virtual and immersive environments, like Virtual Reality and Augmented Reality. Therefore, not only able to be reproduced in one framed two-dimensional angle, but at any angle, on a tridimensional space. Where the spectator have the possibility to walk around the subject, come closer or further and even interact with the subject in a natural manner.

3.1. Wrapping Process

The first process in this workflow is called "Wrapping Process" (R3DS, 2018), in this process the creation of a new topology for the character is made by wrapping a 3D mesh on top of the generated models.

This process is important since the 3D meshes generated in the last step are already low-poly and good to work with, but they don't share the same topology, meaning the same 3D information, therefore isn't possible to morph between the different 3D facial expression models. To be able to morph between one expression to another, all of the 3D meshes must contain the exactly same values in their tridimensional data.

In this case was used a software called R3DS Wrap 3.3 (R3DS, 2018), it is a wrapping software becoming widely used in the games and 3D modelling field.

3.1.1. Alignment and Point Selection

This process starts by importing the Neutral expression 3D mesh generated in the last chapter inside the R3DS Wrap 3.3 software. In here, it's possible to already noticed how the polygons of the generated mesh looks unpractical to manipulate.

In order to fix this, a "Base Mesh" is chosen from the software's gallery, that is a perfect topology mesh created by the software team to be used in this process. It's noticeable how the Basemesh's polygons looks polished and easier of manipulation than our generated mesh from the data-set (Fig. 14).

In this wrapping process, the software basically wraps the Base Mesh on top of our generated scanned mesh, modifying it to match the exactly form of the face, therefore creating a perfect topology version of our scanned 3D character.

The first step is to align the generated mesh on top of the Base Mesh by positioning, scaling and rotating. After this rough alignment, a "Wrapping" node is chosen and connected to the 3D meshes into the correspondent inputs (Fig. 15).

It is noticeable that there's a yellow input in the Wrapping node called "Point Correspondences", this input is used in order to notify the wrapping algorithm which points are correspondence between the Base Mesh and our generated Neutral Mesh. For that, a node called "Select Points" is created and by connecting both the meshes it's possible to create correspondent points between the two.

It's noticeble that there's a yellow input in the Wrapping node called "Point Correspondences", this input is used in order to notify the wrapping algorithm which points are correspondence between the Base Mesh and our generated Neutral Mesh. For that, a node called "Select Points" is created and by connecting both the meshes it's possible to create correspondent points between the two.

As can be noticed by the picture (Fig. 16), the selection happens in the corner of the eyes and mouth to guide the algorithm to create a good wrapping of the Base Mesh on top of our generated mesh. A "Select Polygons" on the Base Mesh can be also applied in order to select the Basemesh mouth socket group, connecting into the last input of the Wrapping node. After this process, the wrapping algorithm is ready to be applied.

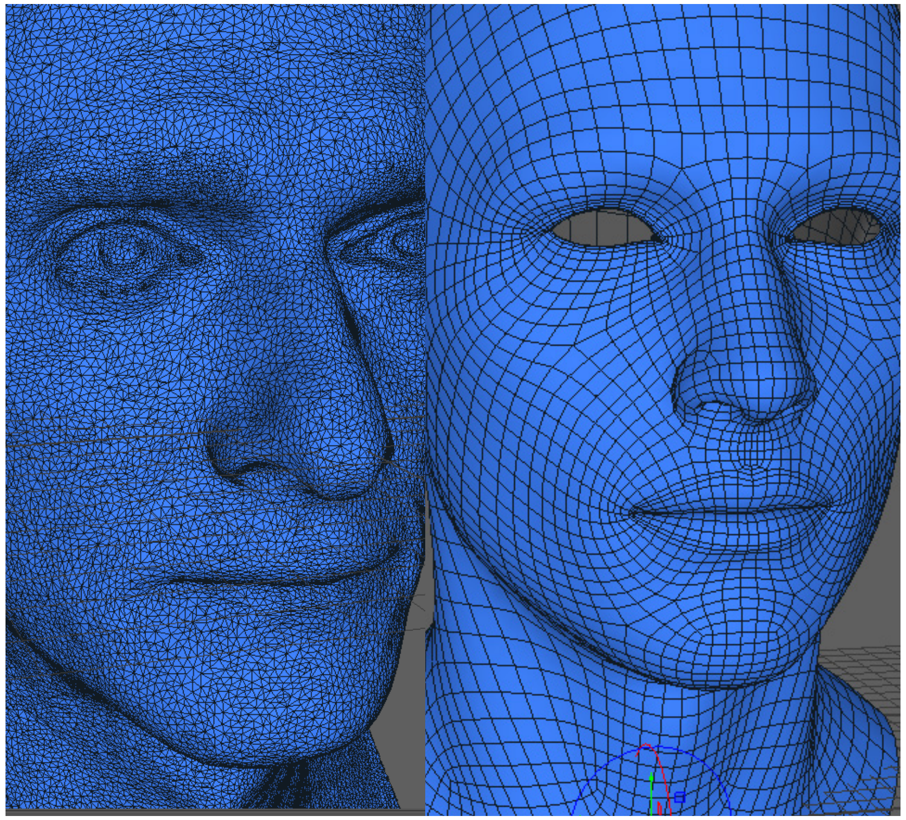


Figure 14: Comparision between mesh topology of generated 3D model and Basemesh.

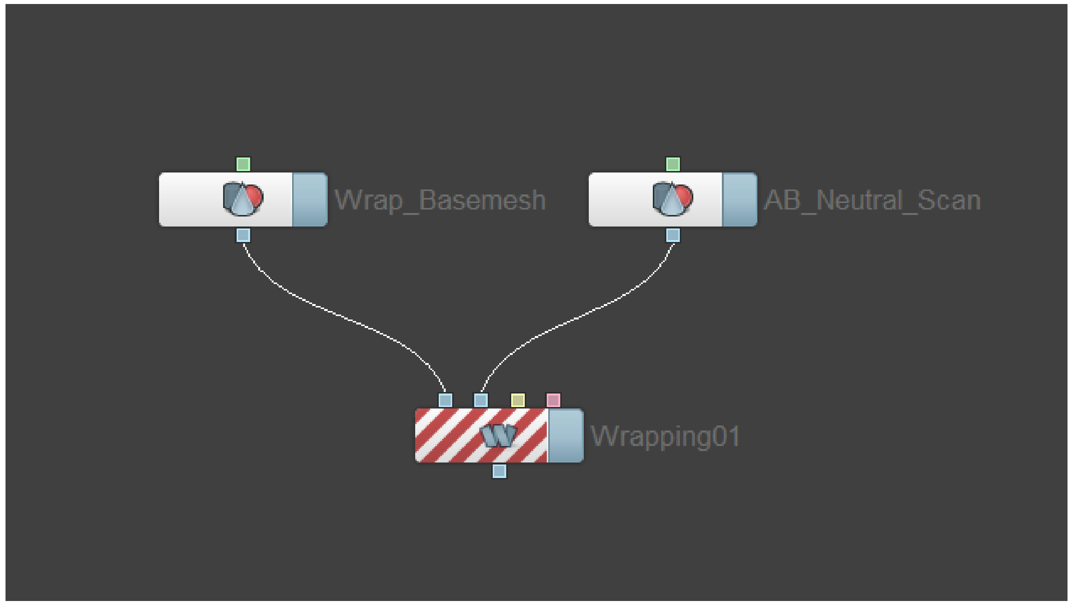


Figure 15: Wrapping Node scheme inside R3DS Wrap software.

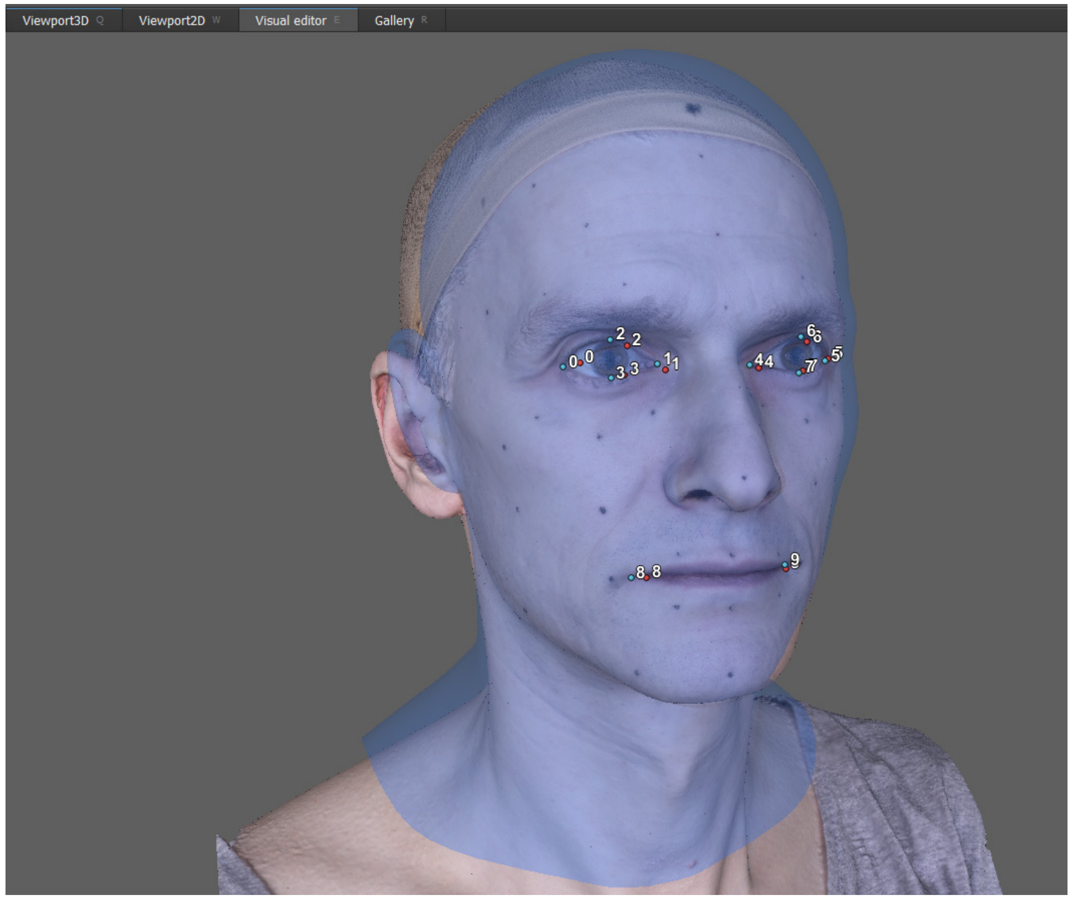


Figure 16: Select Points set to serve as guide for the Wrapping algorithm.

3.1.2. Wrapping

Now that both the software's Basemesh and the Neutral expression generated mesh are aligned and the algorithm has a starting knowledge based on the Point Selection that was made, it's possible to compute the wrapping process. At this stage, it is possible to check in real-time while the computer wraps the Base Mesh and modify it in order to look exactly like the Neutral generated mesh.

When the process is done, a new Neutral expression model with a perfect topology is generated (Fig. 17). In this process it's noticiable that the software also creates a hole were the eyes should be positioned. This is a common practice since the eyes are meant to be implemented after the facial model process, as a separated and independent entity of the character's 3D model.

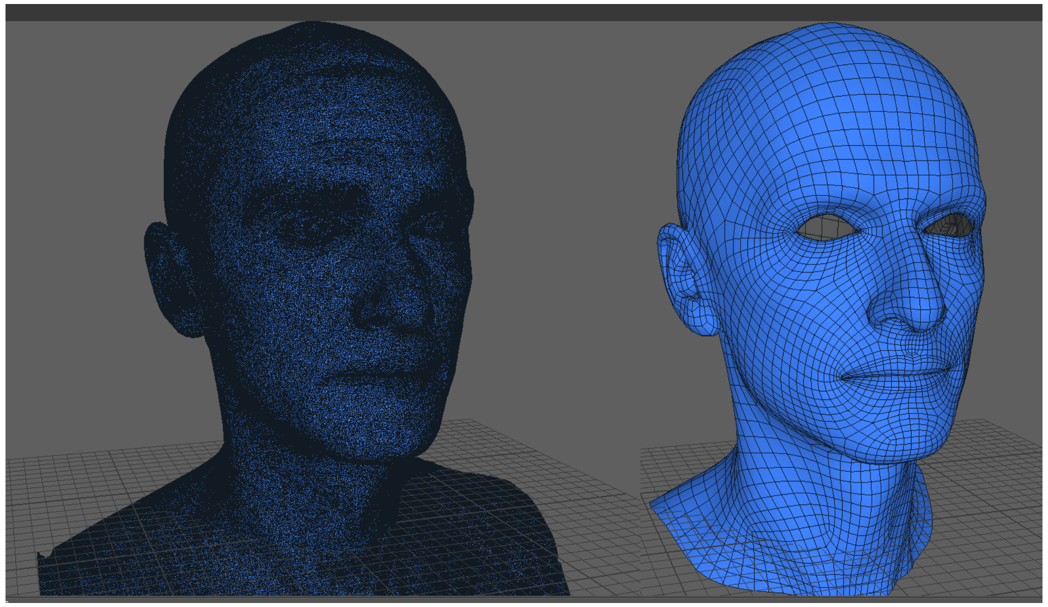


Figure 17: Comparision between complex mesh topology and a perfect topology after wrapping.

3.1.3. Texture Transfer

Now that a perfect topology Neutral 3D mesh of the scanned head is created, the next step is to create the proper texture for this model. Since the vertex information was completely changed in the topology process, if the same texture, from the previous model, is applied to the new one, a totally mistaken result is generated.

Therefore, the next step to be done is the Texture Transfer from the old model to the new retopology. For that, a node called "TransferTexture" is used, in which the old model and the new model serves as inputs. It is also possible to choose to ignore some polygons if needed. The result this process presents is a totally new texture, ready to be applied to the corrected topology Neutral mesh (Fig. 18).

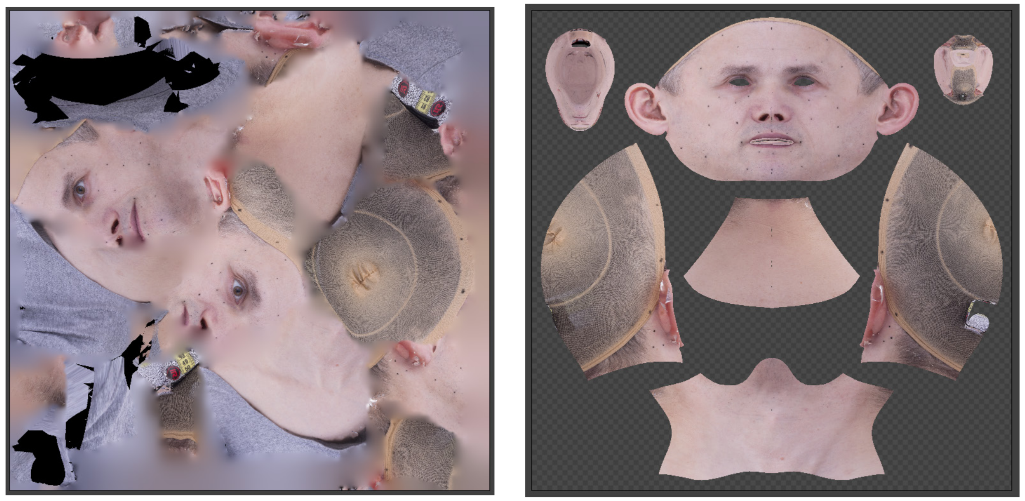


Figure 18: Texture generated to fit the new head mesh topology.

One important detail is that this process can create artifacts on the corners of the eyes and mouth, also, it generates textures into the mouth polygons, which aren't needed. These artifacts can be easily cleaned using Photoshop and saving the image again (Fig. 19).

The last step after the fix of texture artifacts, is to use a node called "Extrapolate Image", this process, has the name indicates, will extrapolate the texture image in order to generate a better result in the corners, avoiding extra artifacts and smooth transitions between sections of the face texture (Fig. 20).

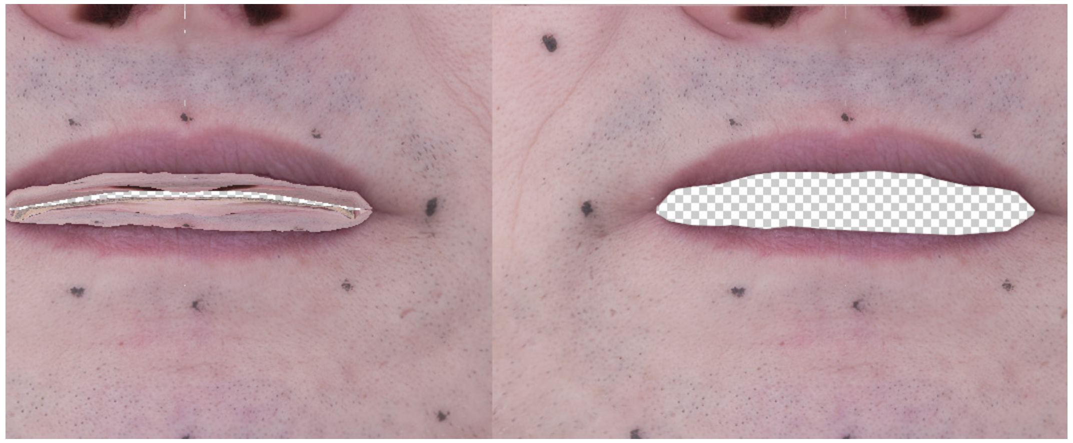


Figure 19: Artifacts cleaning on the corners of the mouth.

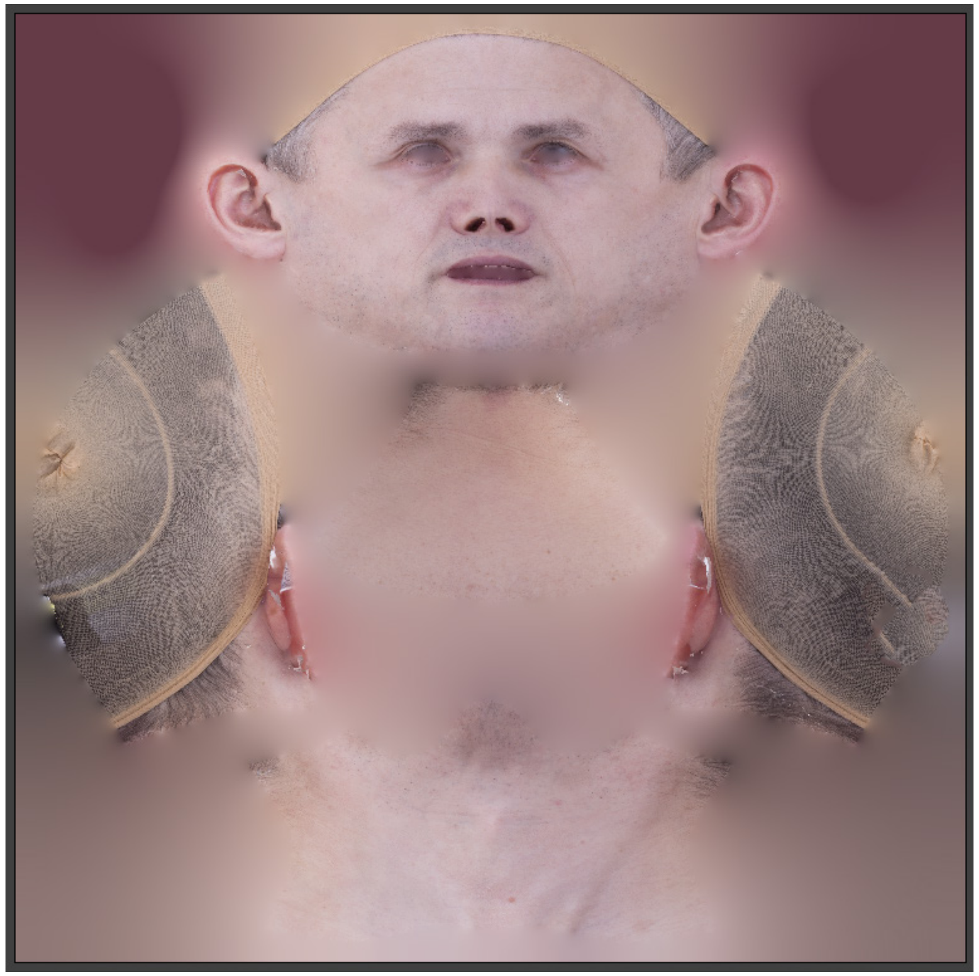


Figure 20: Texture after extrapolation.

3.1.4. Blendshapes and Wrapping Process Conclusions

In this process I demonstrated how to convert a generated 3D mesh from a real person's scan into a topology ready workable 3D asset. This process must be done to all the expressions that are captured.

After all the processes are done, a data-set of 3D meshes expressions that share the same topology are made. Therefore, it is possible to morph between those expressions using a technique called "Blendshape". In this case, it was used Autodesk's 3Ds Max to create a 3D model that can morph between the different expressions, therefore a "blendshape model", this can be done with mostly any 3D editing software in the market, like Maya or Blender.

This process is relatively simple. In order to achieve a blendable model, all the expression models are imported inside a new scene in the software. The Neutral expression model is selected and applied a modifier called "Morpher", this modifier has a list of possible inputs for other 3D meshes that share the same topology. Therefore, it is possible to choose every expression generated and imported as inputs. After that, it's possible to morph between the different meshes and also combine their values, in this case going from a value of 0 to 1 (Fig. 21).

At this point the model is completely ready to be exported and imported into a real-time engine, in which it's possible to morph in real-time to create animations, apply systems to drive the facial expressions and create an immersive experience utilizing a virtual character scanned from a real actor or person that has movement and behaviour.

Within this workflow, it is possible to capture the nuances and movement of a facial tridimensional character by using techniques of volumetric capture, it allow us to rely on realistic data. Opposite to the traditional pipelines, in which a 3D artist must make all the different movements by hand, distorting the mesh, using photographs only has references.

Often, this hand made process doesn't bring satisfactory results once the human face as several little details on muscles that deforms when an expression is made. It is of great challenge for an artist to re-create this small movements by hand sculpting the model. Therefore, the possibility of scanning a real person in such an expression is of great value for our process and workflow.

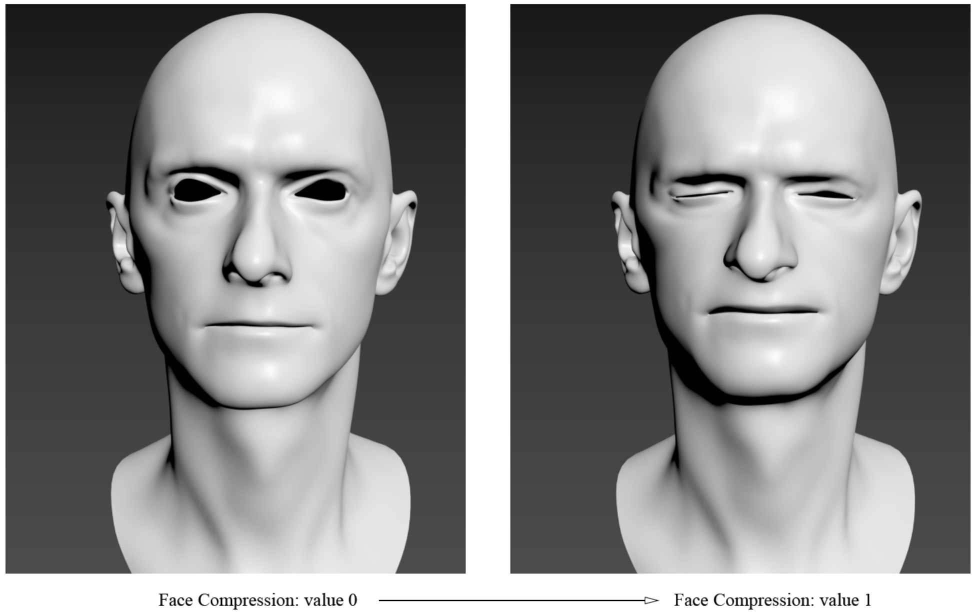


Figure 21: Blendshape morphing between two expressions.

3.2. Importing Generated Virtual Character inside a Real-time engine

In the last sessions I presented steps on how to create a realistic virtual character possible of morphing between different expressions from a series of photographs of a real person. Now that a virtual character consisting of a perfect 3D mesh that contains enough information to be able to morph to different states, and posses a perfect matched texture, it's possible to implement this model into a real-time engine.

In this case, it was used Unreal Engine (Games, 2007) as the game engine of choice, this choice was made in particular for its powerful rendering technology, the ability of implement interactions on a node-based interface and create shaders and material effects in a more suitable way for artists, without much coding background.

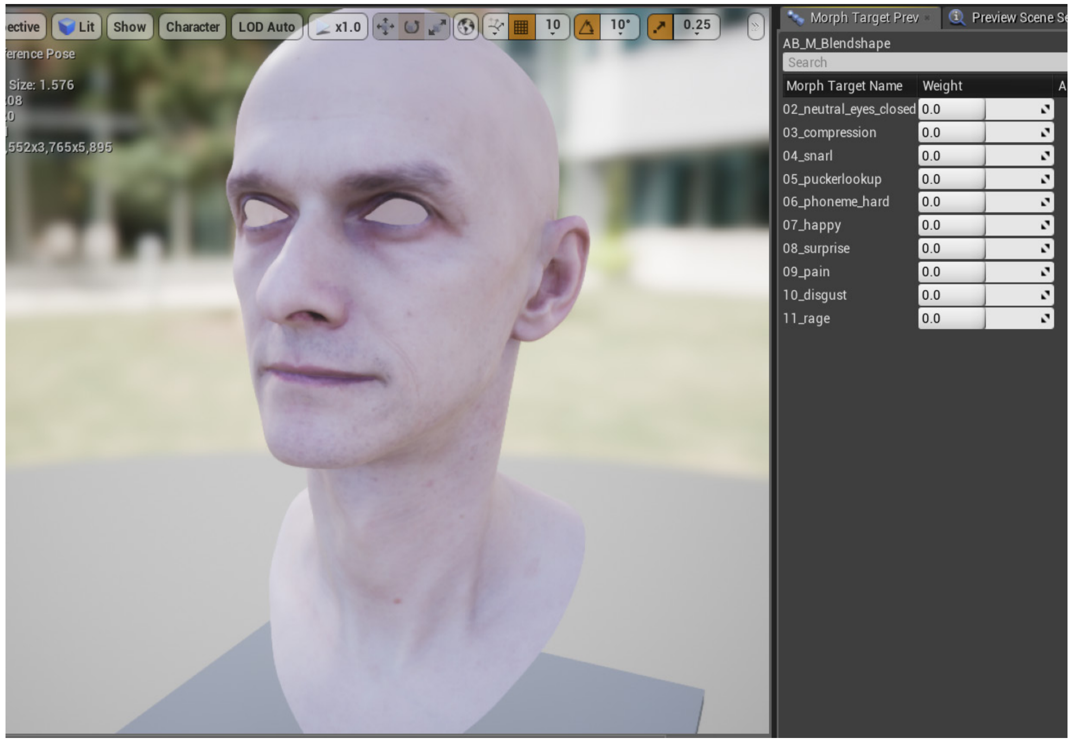
This step starts by the importing of the virtual generated character that contains the Blendshape data (or Morph Targets), and also the corresponding Neutral texture in a PNG format inside the engine environment, in here is important to note that the "Morph Targets" option is selected, otherwise our morph information won't be imported to the engine. By opening our character file inside the engine it's possible to verify that all the Morph Targets are imported correctly and it's possible to change their values in order to modify the character's expression.

At this point, the texture looks in perfect place but lacks information on small details, in which the light bounces and creates a greater sense of depth, such as skin wrinkles, corner of the eyes and other facial details (Fig. 22). To enhance these details, it's possible to create a Normal Map from the texture, a Normal Map is a common technique in computer graphics in which pixel colour is used as information of depth, creating small 3D details automatically without the need of modelling by hand (Xia et al., 2011).

In this case, a software called "xNormal" (Orgaz, 2010) was used, which can generate a Normal Map based on a height-map of a texture. Therefore, a grayscale version of the Neutral character texture is imported, in which a black pixel means it is further from the view and a white pixel means it is closer. With this information, the software can generate a Normal Map automatically that can be applied to our character inside the real-time engine.

After applying the generated Normal Map in the avatar, it's possible to check closer to the model and verify the difference it makes for a greater aesthetical and realistic look, especially regarding on how the virtual lights bounces when lightning the virtual being.

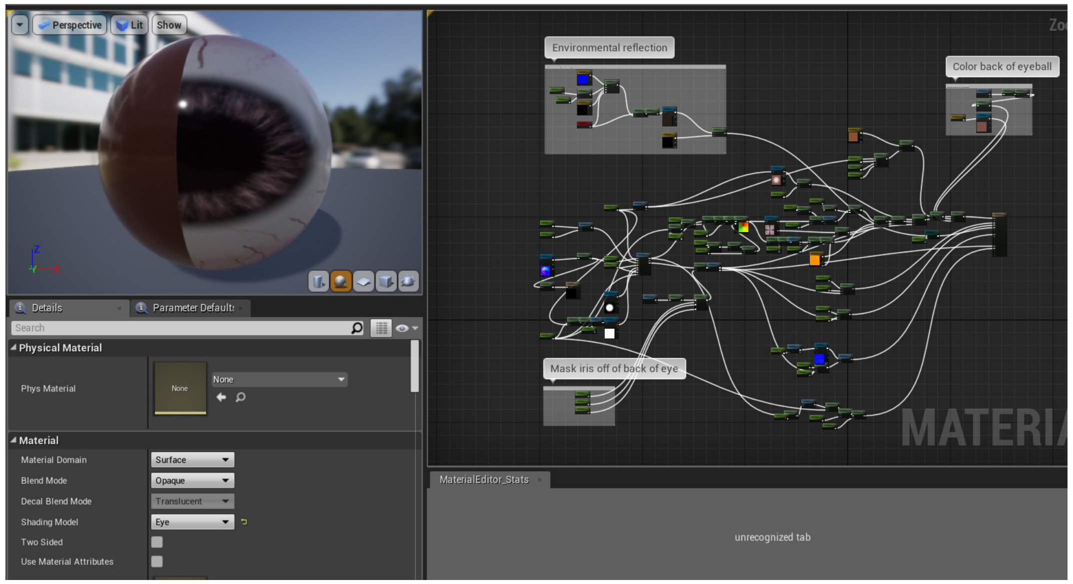
Advanced techniques such as Subsurface Scattering and custom shader material authoring, are highly specific topics in computer graphics and will not be covered in this thesis work. The development and achievements of such a field is of great importance for the enhancement of realism in virtual characters. Although, for the project of this thesis, the team didn't feel the necessity of diving in such complex and advanced technologies at this point, these will be researched in future projects.



AB_M_Blendshape

Morph Target Name	Weight
02_neutral_eyes_closed	0.0
03_compression	0.0
04_snarl	0.0
05_puckerlookup	0.0
06_phoneme_hard	0.0
07_happy	0.0
08_surprise	0.0
09_pain	0.0
10_disgust	0.0
11_rage	0.0

Figure 22: Character inside Unreal Engine real-time environment.



Physical Material

Phys Material	None

Material

Material Domain	Surface
Blend Mode	Opaque
Decal Blend Mode	Translucent
Shading Model	Eye
Two Sided	□
Use Material Attributes	□

Figure 23: Unreal Engine's eye template shader network.

3.2.1. Eyes Implementation

The implementation of eyes for the character was made inside Unreal Engine environment. As it is common on a real-engine framework, many examples, templates and lessons are shared by the community in order to speed-up processes. In the case of the implementation of eyes to a virtual character, Unreal Engine's provide a really good example to be used on its user's projects.

Virtual representation of eyes is an extremely complicated subject that needs very specific knowledge, teams and researchers to create a believable piece. Unreal Engine's research and technical team created a sample to be used by artists, research teams and studios in order to avoid the need to dive into this complex subject (Fig. 23).

Therefore, it was chosen to use Unreal Engine's ready made eye representation solution to be implemented on our virtual character, which happened to be a perfect match and allow to create eye interactions in order to enhance the virtual experiences made with the character.

3.2.2. Teeth, Tongue and Gum Implementation

In the case of teeth, tongue and gum, it is not possible to create a 3D mesh of of these features using the same process to create the facial expressions. Therefore, in this case, a traditionally modelled character's mouth was created based on the photographic reference. After it is modelled, it's possible to import inside the real-time engine and synchronize these elements to the character's head and face. On this particular character, the main 3D modeller of the project modelled the teeth and gum by hand modelling.

3.3. Wrapping and Importing Conclusions

In this chapter I presented a workflow for the creation of a real-time ready virtual character based on the capture of a real actor using a photogrammetry setup. This process is substantially faster then the traditional hand modelling and hand-work methods, without the advance of automated and algorithm based software and processes.

It also presents with a similar and many times better results from a traditional workflow, since everything is being captured from a real subject, instead of artificially done using real photographs only as references.

With the character ready inside a real-time engine and being able to perform facial expressions by morphing between the different poses, it gives the possibility to implement all the nuances a real-time environment makes possible. Real-time lightning, animation, interactions and artificial intelligence systems can now be applied to our character.

In the next section, I will demonstrate how to connect the morphing system of our character to an Open Sound Control (OSC) (Schmeder, Freed, & Wessel, 2010) connection in order to input data from a Networked system, such as an Artificial Intelligence system based on Convolutional Neural Network and Machine learning application to drive the character's expressions, and also accessible ways for facial motion capture animation.

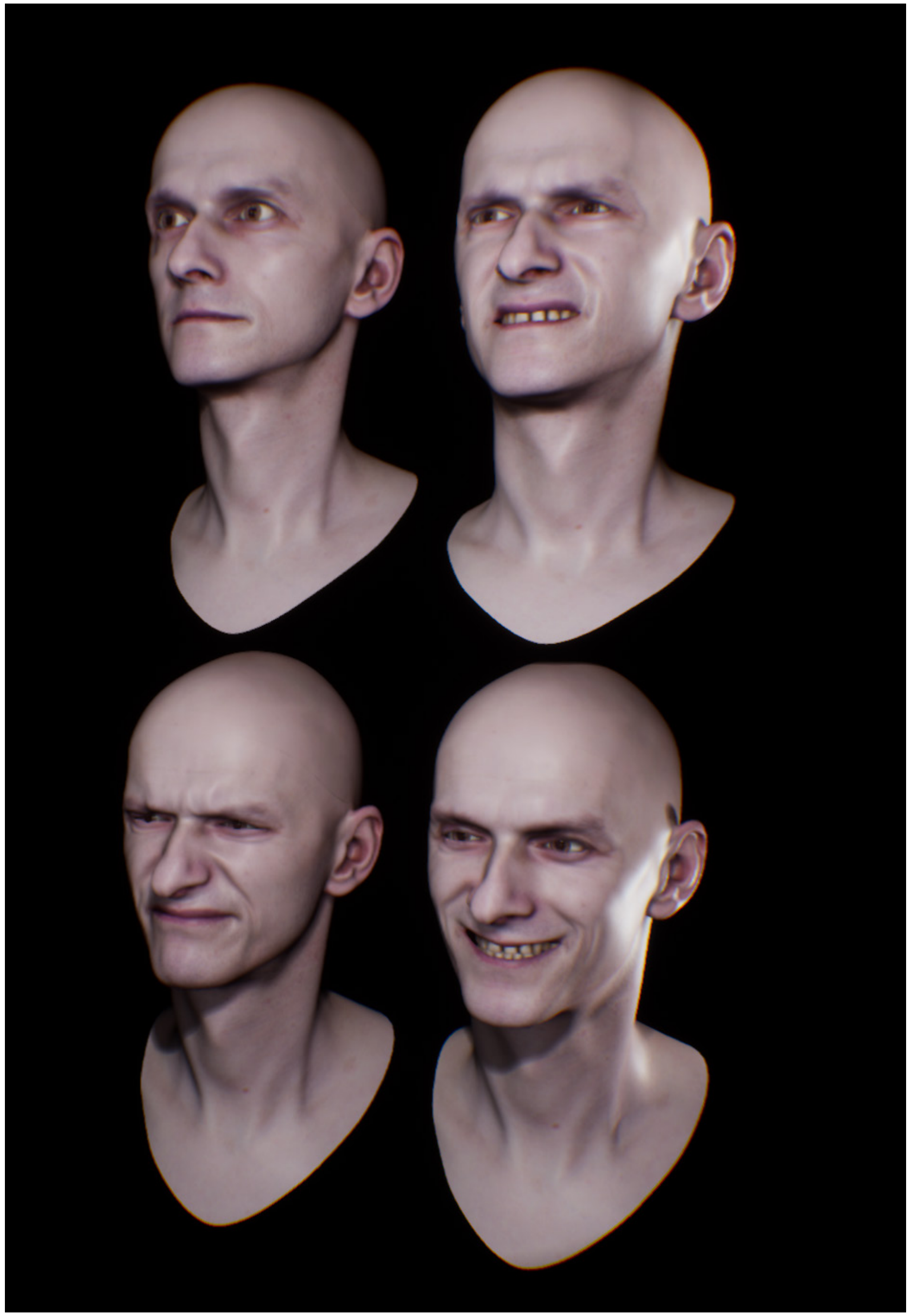


Figure 24: Virtual Character expressing different facial reactions.

4. Discussion: Possibilities for Real-time Interactions with a Volumetric Captured Virtual Character

In the previous chapters thesis I have described a specific workflow for the creation of a real-time virtual character based on capturing techniques. Once the character is imported inside such an environment, a range of possibilities for real-time interactions emerge. Since we have access to the different expressions morphing data of the different facial expressions. In this chapter I will discuss these real-time interaction possibilities in more detail. Furthermore, I will also discuss the three practical experiments that were created with the aim of driving the virtual character's facial expressions in real-time.

For the he experiments I had set up the following tasks: (1) to position our virtual character into the Virtual Reality environment by taking advantage of the real-time lighting and rendering power of Unreal Engine (2) to view the character in a complete immersive scenario and implement real-time data exchange between applications to drive the facial expressions and (3) to apply accessible facial motion capture in real-time.

4.1. Experiment in a Virtual Reality Environment

With the first experiment, the character model turned out visually impressive in terms of facial features and humanlikeness. These features have been identified as one of our biggest challenges. This experiment allowed further analysis and critical discussion on particular aspects of the 3D model itself. One of these aspects related to the lack of wrinkle details on the model. Although the 3D model uses a normal map with depth information, it is not enough to fully replicate all wrinkles and small facial hair features that are characteristic for a human face. To be able to reproduce these human skin details to the 3D model, advanced technologies are needed, such as a set of detailed shading techniques as well as a set of advanced graphics solutions for enhancing the texture of the character. This cannot be achieved within the current state-of-the-art of simple texturing systems, without introducing advanced techniques and a heavy manual labour.

4.2. Experiment Applying Open Sound Control

The second experiment was designed to implement Open Sound Control (OSC) signal in order to control the facial expressions of the character. OSC is a format for stream of real-time control messages (Schmeder, Freed, & Wessel, 2010). Although, as the name indicates, it is mainly used for audio signal exchange, since it shares data in an easy way over network, it has been used for several applications, not limited to audio (Schmeder et al., 2009).

For this work, I created an OSC system using Unreal Engine's Blueprint programming system, based on the plugin by Guillaume Buisson (2014), which enabled the control of the character's expressions over network (Fig. 25).

To apply OSC signals to control the character's expressions, Machine Learning principles for the facial movement were implemented. The application FaceOSC (McDonald, 2012) allowed us to transfer facial features data of a viewer that is positioned in front of the webcam, to the 3D character within the game engine. The facial features of the viewer are sent via OSC signals,

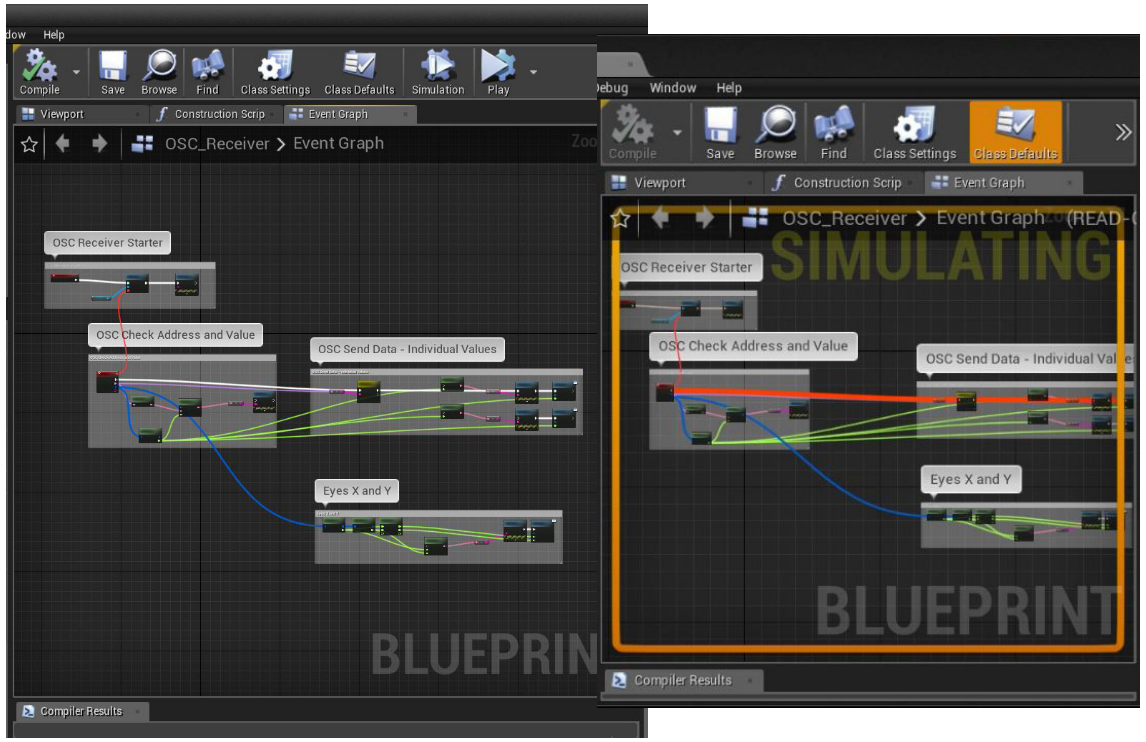
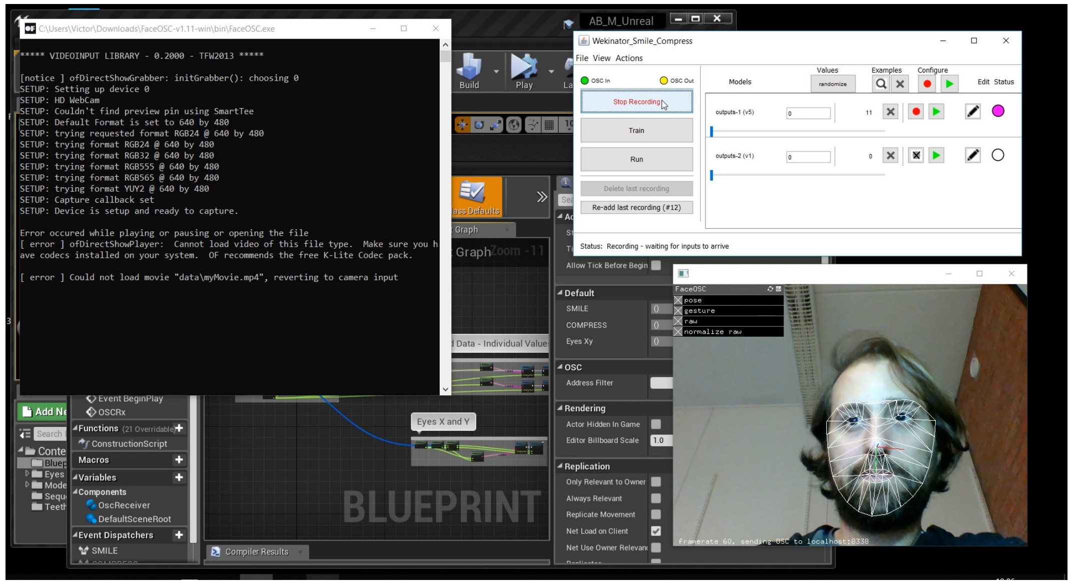


Figure 25: Open Sound Control system inside Unreal Engine environment.


C:\Users\Victor\Downloads\FaceOSC-v1.11-win\bin\FaceOSC.exe

```
***** VIDEOINPUT LIBRARY - 0.2000 - TFW2013 *****

[notice ] ofDirectShowGrabber: initGrabber(): choosing 0
SETUP: Setting up device 0
SETUP: HD WebCam
SETUP: Couldn't find preview pin using SmartTee
SETUP: Default Format is set to 640 by 480
SETUP: trying requested format RGB24 @ 640 by 480
SETUP: trying format RGB24 @ 640 by 480
SETUP: trying format RGB32 @ 640 by 480
SETUP: trying format RGB555 @ 640 by 480
SETUP: trying format RGB565 @ 640 by 480
SETUP: trying format YUY2 @ 640 by 480
SETUP: Capture callback set
SETUP: Device is setup and ready to capture.

Error occured while playing or pausing or opening the file
[ error ] ofDirectShowPlayer:  Cannot load video of this file type.  Make sure you h
ave codecs installed on your system.  OF recommends the free K-Lite Codec pack.

[ error ] Could not load movie "data\myMovie.mp4", reverting to camera input
```

Wekinator_Smile_Compress

Status: Recording - waiting for inputs to arrive

Figure 26: Experiment using Machine Learning application to drive character's facial features.

that are changing in real-time according to the viewer's facial expressions and movements.

After the data of the facial behaviour of the viewer has been received to the game engine, the next step is to use a Machine Learning application called Wekinator (Fiebrink & Cook, 2010). This application allows us to train the 3D character's facial expressions based on the recorded data of human dynamical expressions. Consider, for example, a neutral facial expression data. The Wekinator application will gather samples of data values that represent a human neutral expression, thus producing an estimate of values that can be categorized as a neutral expression (Fig. 26).

Similarly, recording smiling facial expressions, the accumulated sample data will be categorized to constitute this particular facial expression. Same with the sad facial expressions, and so on. By this training method, the application is capable to learn which range of data represents a neutral expression and which range represents a smiling, or alternatively, sad expression. After the training is done, the application runs the data, that can be controlled using facial expressions from the user via the computer's webcam.

The viewer's facial expression data can then be sent again, via OSC, back to the virtual character inside the real-time environment. And one can control the virtual character's facial behaviour accordingly to the trained data from Wekinator, putting in simple words, as of the example mentioned, we can teach the virtual character how to smile.

4.3. Experiment with Facial Motion Capture

The third experiment focused on the usage of OSC signals to drive the virtual character using possible Facial Motion Capture solutions. Motion Capture techniques are known to be expensive and to require specific knowledge, fully equipped studio and trained teams to give good results (Balakrishnan & Diefenbach, 2013).

However, after the launch of the iPhone X device (Apple, 2018), which contains hardware capable of creating facial capture using the frontal camera of the phone, developers started to experiment with more accessible ways for capturing facial performances using the device (Strassburger, 2018).

At this phase, the principle is to add more blendshape information to the virtual character, accordingly to Apple (2018) guideline and, in the case of this thesis work, created by the 3D artist Eeva R. Tikka (2018). We recorded facial performance using the device's frontal facial camera technology, using the application FaceCap (Jansson, 2018). In FaceCap it is possible to record a facial performance and export as a base 3D model containing the facial features data. After this recording, the data can be tranferred from the application's 3D model to our own experiment's virtual character.

The result is an animation ready to be used inside the real-time engine, without the need of high-end and expensive facial motion capture hardware, which has been one of the main motivations for the current thesis work.

5. Evaluation: Possibilities for Real-time Interactions with a Volumetric Captured Virtual Character

In this thesis I have shown, that, based on volumetric capturing techniques and the described workflow, the creation of a virtual character can be conducted and reproduced with accessible low-cost hardware and software and without extensive manual animation work. In other words, one can achieve a real-time action-ready character without the need of heavy hand work and expensive investment to high-end studio resources. It proved to be a framework possible to be used by individuals, small and independent teams. It is also valuable for experiments, proof-of-concepts, and rapid prototyping.

This process however, isn't ready for high-end virtual characters in projects that seek for high perceptual realism in terms of human facial behaviour or body movements. The models presented in this thesis look relatively realistic in terms of human-likeness. However, details such as wrinkles, provided typically by advanced computer graphics techniques, subsurface scattering and other shading systems, can't be yet achieved by this mostly automated workflow.

In the presented thesis work, implementing the OSC connection to exchange data in real-time between distinct applications proved to be of a great value. With this possibility, one is able to create applications and experiments to drive the character's facial expressions in real time. Since it is a system for data exchange, several applications and systems can be created outside the real-time environment but still connect to the virtual character data. As for example, one could send signals from a musical instrument to the character's facial data in order to drive the facial expressions by playing musical notes.

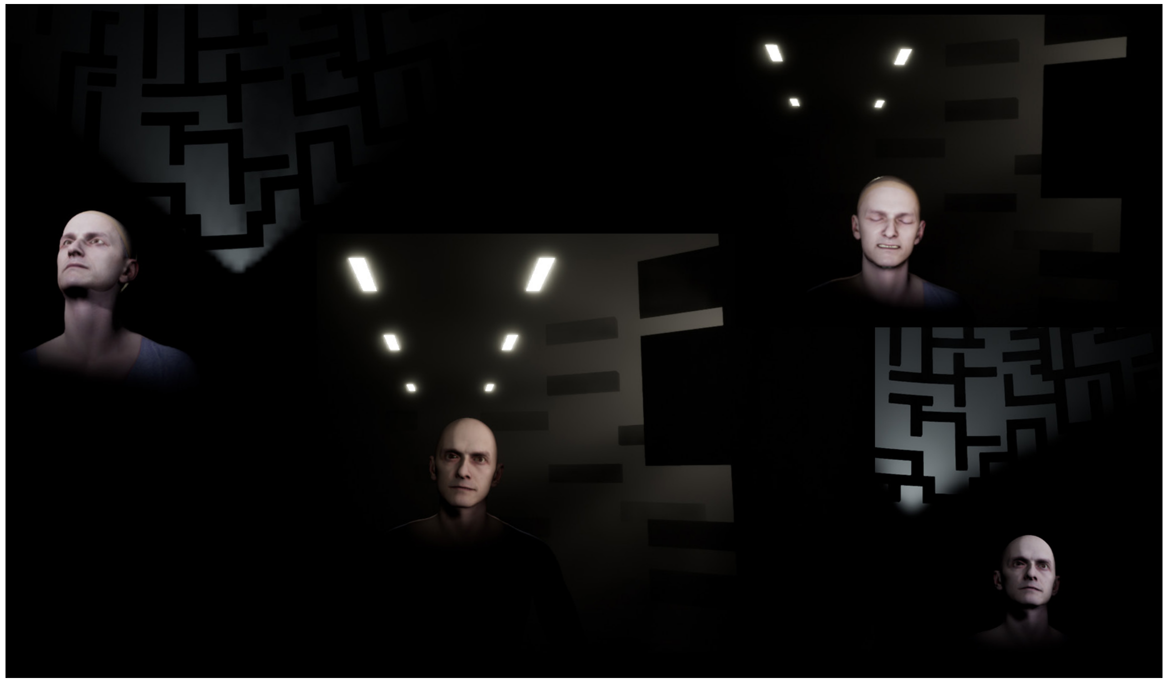


Figure 27: The State of Darkness VR experience.

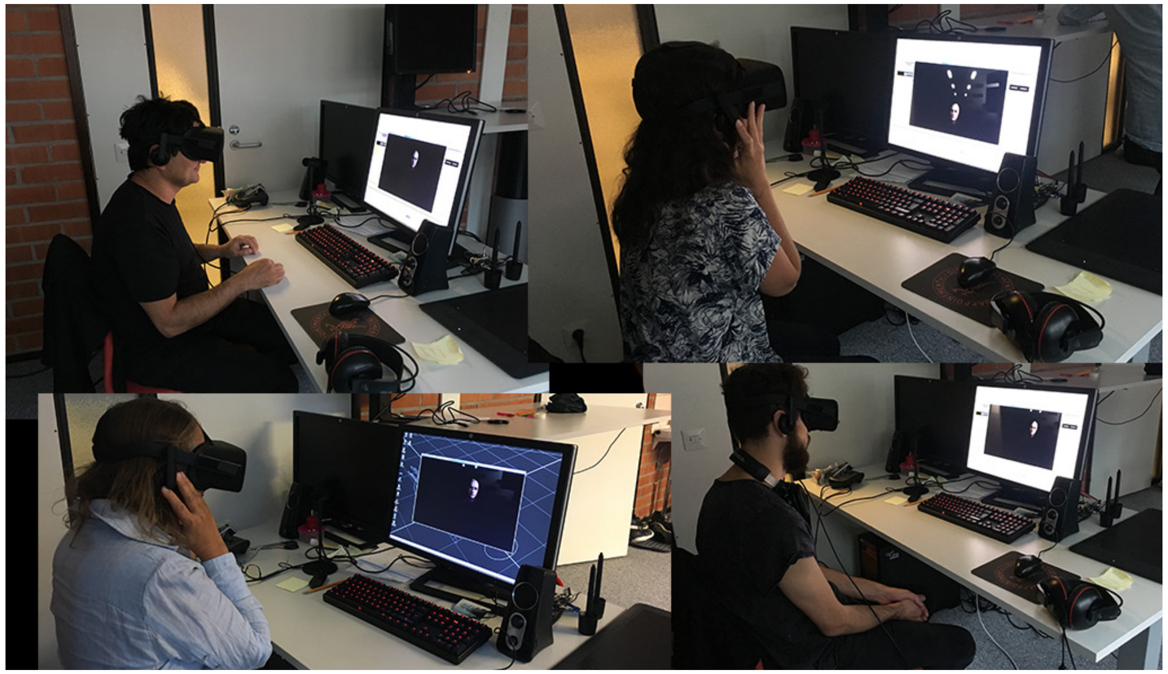


Figure 28: Participants experiencing The State of Darkness.

With this possibility, Artificial Intelligence systems which exchange data through network can be created in order to drive the character's facial expressions based on Neural Networks or Machine Learning applications. This also opens possibility to interaction with user's biofeedback and unconscious interactions, which is being already experimented in the two enactive virtual character installations The State of Darkness and The Booth (Tikka et al., 2018; See Figures 27 and 29) In these art installations each viewer's encounter with a virtual 3D character in immersive environment is tracked by biosensors, and the resulting biodata is fed back to the narrative system. This biofeedback drives the character's behaviours and facial responses in a continuous looping manner. In short, collecting real-time biofeedback from the viewer in order to drive the character's facial reaction accordingly to these physiological responses.

Within this framework, was also shown that it can be applied to new accessible facial motion capture possibilities. With a single device such like the iPhone X, one may drive the virtual character and record facial animations or even real-time driven performances. The recordings can then be used in a range of applications such as games, films and art installations. This technique can also be mixed with OSC signals coming from an outside the applications, therefore, even if there is a base animation driving the character, it can still contain emergent real-time modifications from OSC data.

This possibility is of great value for experiments in which the interaction of the viewer must be mixed with a base animation in order to keep a realistic and concise interaction feedback

6. Outlook to the future.

The results of the work I have presented in this thesis, opens possibilities for future research. The processes and workflows developed and discussed here have focused on automation a thus avoiding heavy hand labour. As such, my work can be characterized as a practice-based research on non-traditional workflows for virtual character creation. However, to reach the humanlikeness and naturalistic look in terms of realism, some details still need to be implemented in a traditional way, such as facial modelling details and shading systems. In the future research, such specific, yet automated techniques must be developed and tested in order to further expand the current framework.

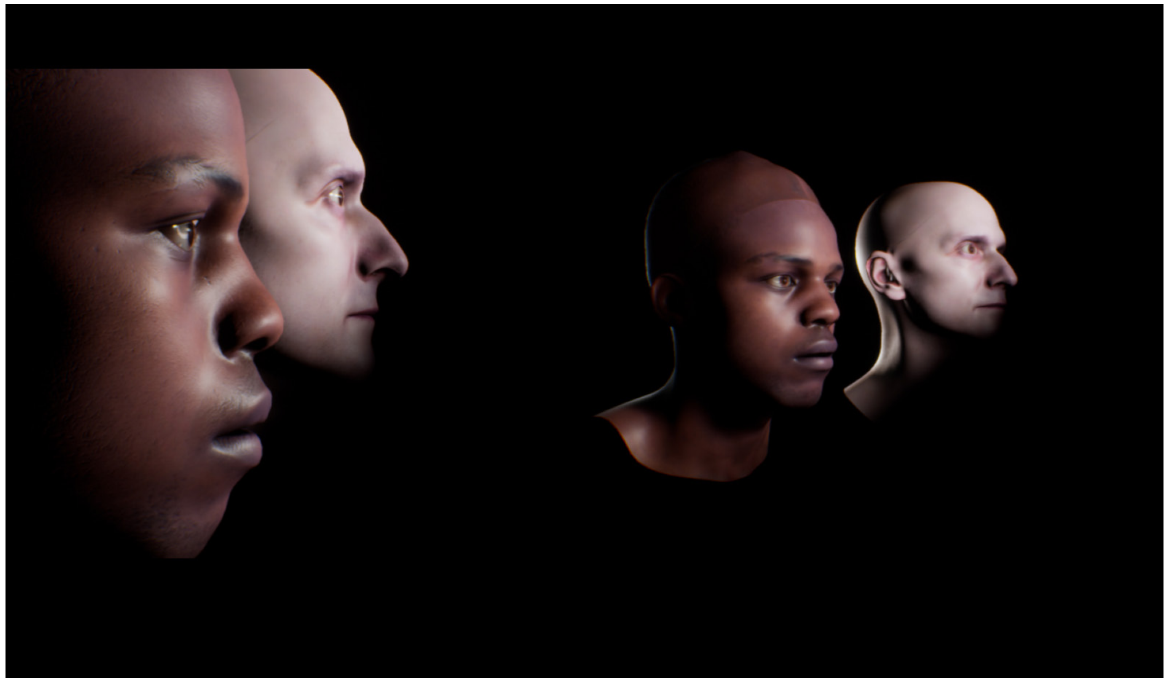


Figure 29: The Booth experience virtual characters.

When looking into the future developments, from the current point of time, new volumetric techniques are being created, one in particular, named Volumetric Video (Collet et al., 2015), consist of the photogrammetrical capture of a subject in a frame-by-frame basis, capturing both tridimensional data and rgb values. This technique is getting increasing attention for its realism compared to virtual mimic of human gestures (Fig. 30).

I'm suggesting that in the future research, one should apply techniques of the framework presented here to the production of a virtual character that is captured via frame-by-frame photogrammetry, or volumetric video capture. Some works needs also to be invested in obtaining captured movements from reality and bringing emergent behaviour and engaging interactions to this sequential capturing technique.

As discussed, the possibility to connect the real-time character to outside applications via network, opens up following research in applying advanced Artificial Intelligence techniques to drive the character's facial reaction. With the advance of Neural Networks and Machine Learning applied to the understanding, training and experiment with human face expressions. The framework that I have presented in this thesis can be of great value for researchers, teams and groups that seek to implement their AI systems into a realistic looking virtual characters running in a real-time engine. This can be used by a range of experiments ranging from AI to biofeedback and physiological computing. Both for scientific experiments as for creative installations.

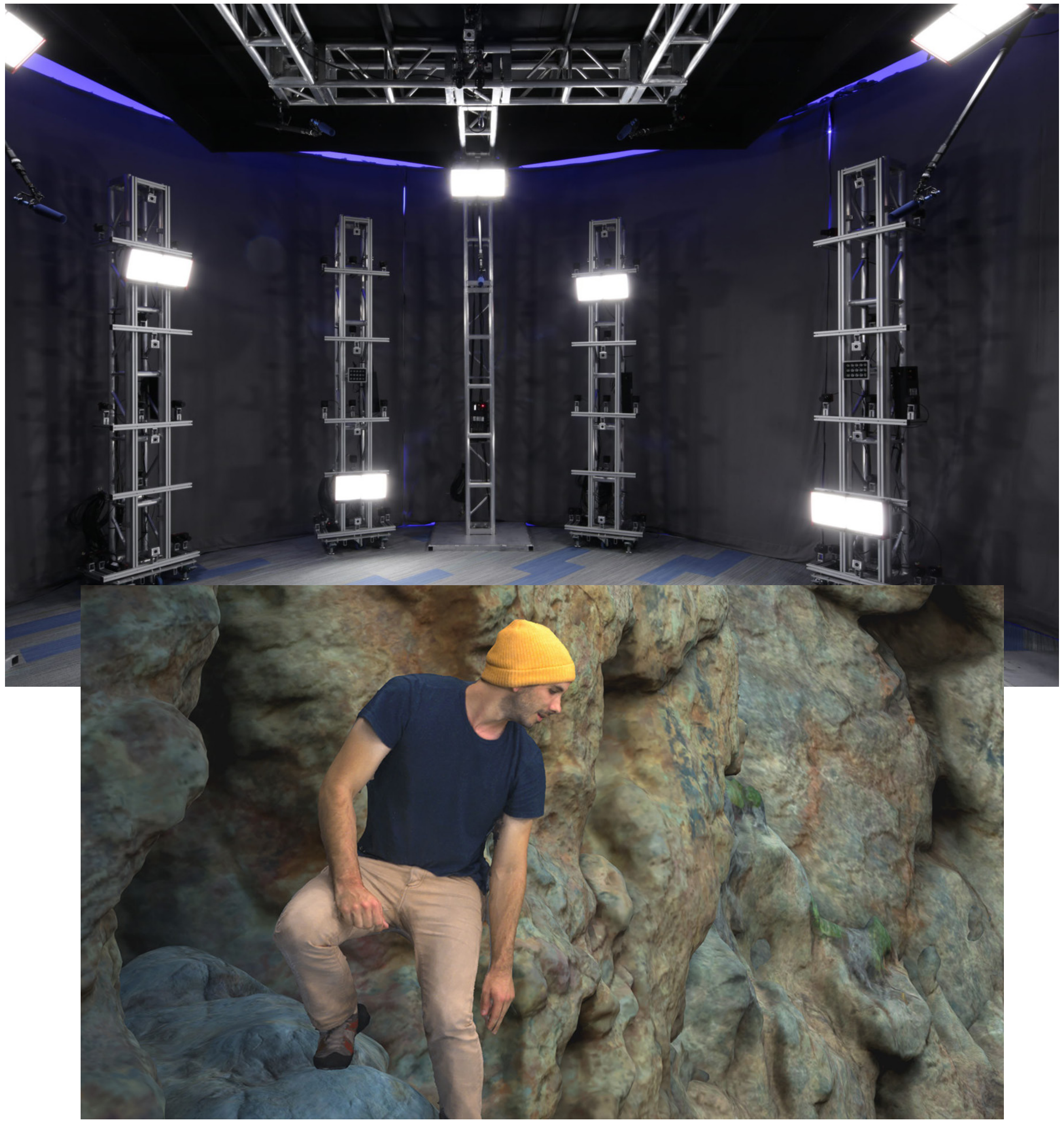


Figure 30: Volumetric Video capture rig (Dimension, 2018) and virtual character (8i, 2018).

7. Conclusions

In this thesis work I presented a complete framework for creating a virtual character based on photogrammetry capturing techniques. I presented several workflows from a capturing phase, in which I used photogrammetry technique to capture a real person.

The processes are taken to generate tridimensional data and make it ready for a real-time environment, where the data can be morphed between different facial expressions. The possibility of manipulation of the facial expression data using real-time applications and techniques under different experiments in order to bring a dynamic behaviour to the virtual character was also shown.

The framework was based on accessible hardware and software, that can be replicated by individuals and small teams, without the need of high budget, expensive equipment and specific knowledge of 3D techniques and software.

With the result of this workflow related to a data-driven character in real-time environment, I envision that my work opens possibilities for a full range of interactive experiments, art installations and immersive real-time productions. It is true especially with those that are based on the relations between the viewer and the virtual being.

6. References

8i. 2018. 8i Volumetric Video Capture. Retrieved from: https://8i.com/

Agisoft, L. L. C. (2014). Agisoft PhotoScan user manual: professional edition.

Alexander, O., Rogers, M., Lambeth, W., Chiang, J. Y., Ma, W. C., Wang, C. C., & Debevec, P. (2010). The digital emily project: Achieving a photorealistic digital actor. IEEE Computer Graphics and Applications, 30(4), 20-31.

Apple. 2017. ARFaceAnchor.BlendShapeLocation. Retrieved from https://developer.apple.com/documentation/arkit/arfaceanchor/blendshapelocation

Balakrishnan, G., & Diefenbach, P. (2013). Virtual cinematography: beyond big studio production. ACM.

Boehler, W., & Marbs, A. (2002). 3D scanning instruments. Proceedings of the CIPA WG, 6, 9-18.

Buisson, G. 2014. UE4-OSC. Retrieved from https://github.com/monsieurgustav/UE4-OSC

Cavazza, M., Charles, F., & Mead, S. J. (2002, July). Interacting with virtual characters in interactive storytelling. In Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1 (pp. 318-325). ACM.

Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., ... & Sullivan, S. (2015). High-quality streamable free-viewpoint video. ACM Transactions on Graphics (TOG), 34(4), 69.

de Borst, A. W., & de Gelder, B. (2015). Is it the real deal? Perception of virtual characters versus humans: an affective cognitive neuroscience perspective. Frontiers in Psychology, 6, 576.

Dimension. 2018. Dimension Studio. Retrieved from: https://www.dimensionstudio.co

Doane, M. A. (2003). The close-up: scale and detail in the cinema. Differences: A Journal of Feminist Cultural Studies, 14(3), 89-111.

Dou, M., Davidson, P., Fanello, S. R., Khamis, S., Kowdle, A., Rhemann, C., ... & Izadi, S. (2017). Motion2fusion: real-time volumetric performance capture. ACM Transactions on Graphics (TOG), 36(6), 246.

Ekman, P., & Rosenberg, E. L. (Eds.). (1997). What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS). Oxford University Press, USA.

Fiebrink, R., & Cook, P. R. (2010). The Wekinator: a system for real-time, interactive machine learning in music. In Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)(Utrecht).

Fleisher, O., & Anlen, S. (2018, August). Volume: 3D reconstruction of history for immersive platforms. In ACM SIGGRAPH 2018 Posters (p. 54). ACM.

Fraunhofer Society (2018). VoluCap. Retrieved from: http://www.volucap.de/

Epic Games (2007). Unreal engine. Online: https://www. unrealengine. com.Proin nec, semper orci. Aliquam erat volutpat. *Morbi non erat et elit facilisis volutpat* (s. 12-13) (2011).

Epic Games (2018). Epic Games and 3Lateral Introduce Digital Andy Serkis. Retrieved from: https://www.unrealengine.com/en-US/blog/epic-games-and-3lateral-introduce-digital-andy-serkis.

George, J. (2012). Spectacle of Change. Retrieved from: http://jamesgeorge.org/Spectacle-of-Change-2012

Graham, P., Fyffe, G., Tonwattanapong, B., Ghosh, A., & Debevec, P. (2015, July). Near-instant capture of high-resolution facial geometry and reflectance. In ACM SIGGRAPH 2015 Talks (p. 32). ACM.

Jansson. N. 2018. Face Cap. Retrieved from https://itunes.apple.com/us/app/face-cap/id1373155478?mt=8

Kaipainen, M., Ravaja, N., Tikka, P., Vuori, R., Pugliese, R., Rapino, M., & Takala, T. (2011). Enactive systems and enactive media: embodied human-machine coupling beyond interfaces. Leonardo, 44(5), 433-438.

Kalra, P., Magnenat-Thalmann, N., Moccozet, L., Sannier, G., Aubel, A., & Thalmann, D. (1998). Real-time animation of realistic virtual humans. IEEE Computer Graphics and Applications, 18(5), 42-56.

Kobbelt, L., Campagna, S., & Seidel, H. P. (1998, June). A general framework for mesh decimation. In Graphics interface (Vol. 98, pp. 43-50).

Koutsoudis, A., Vidmar, B., Ioannakis, G., Arnaoutoglou, F., Pavlidis, G., & Chamzas, C. (2014). Multi-image 3D reconstruction data evaluation. Journal of Cultural Heritage, 15(1), 73-79.

Kätsyri, J., Förger, K., Mäkäräinen, M., & Takala, T. (2015). A review of empirical evidence on different uncanny valley hypotheses: support for perceptual mismatch as one road to the valley of eeriness. Frontiers in psychology, 6, 390.

Luhmann, T., Robson, S., Kyle, S. A., & Harley, I. A. (2006). Close range photogrammetry: principles, techniques and applications. Whittles.

McDonald, P. (1998). Film acting.

McDonald, K. (2012). Faceosc.

McDonnell, R., Breidt, M., & Bülthoff, H. H. (2012). Render me real?: investigating the effect of render style on the perception of animated virtual humans. ACM Transactions on Graphics (TOG), 31(4), 91.

Microsoft. 2018. Mixed Reality Capture Studio. Retrieved from: https://www.microsoft.com/en-us/mixed-reality/capture-studios

Monaco, J., & Lindroth, D. (2000). How to read a film: the world of movies, media, and multimedia: language, history, theory. Oxford University Press, USA.

ORGAZ, S., 2010. Xnormal. Retrieved from: http://www.xnormal.net/.

Parke, F. I. (1972, August). Computer generated animation of faces. In Proceedings of the ACM annual conference-Volume 1 (pp. 451-457). ACM.

Pighin, F., Szeliski, R., & Salesin, D. H. (2002). Modeling and animating realistic faces from images. International Journal of Computer Vision, 50(2), 143-169.

Poznanski, A. (2014). Visual Revolution of The Vanishing of Ethan Carter. Retrieved from: http://www.theastronauts.com/2014/03/visual-revolution-vanishing-ethan-carter/

R3DS. 2018. Retrieved from: https://www.russian3dscanner.com/docs/Wrap3/index.html

Ruan, G., Wernert, E., Gniady, T., Tuna, E., & Sherman, W. (2018, July). High Performance Photogrammetry for Academic Research. In Proceedings of the Practice and Experience on Advanced Research Computing (p. 45). ACM.

Sagar, M., Seymour, M., & Henderson, A. (2016). Creating connection with autonomous facial animation. Communications of the ACM, 59(12), 82-91.

Sarbolandi, H.; Lefloch, D.; Kolb, A. Kinect Range Sensing: Structured-Light versus Time-of-Flight Kinect. Comput. Vis. Image Underst. 2015, 139, 1–20.

Scatter (2017). Depthkit, Visualize. Retrieved from: http://www.depthkit.tv/visualize/

Schmeder, A., Freed, A., & Wessel, D. (2010, May). Best practices for open sound control. In Linux Audio Conference (Vol. 10).

Statham, Nataska. "Use of Photogrammetry in Video Games: A Historical Overview." Games and Culture (2018): 1555412018786415.

Strassburger, C. 2018. Democratising Mocap: Real-Time Full-Performance Motion Capture with an iPhone X, Xsens, IKINEMA, and Unreal Engine. Siggraph 2018, Real-Time Live!

Tilbury, Richard (2012). Interview with Infinite-Realities. Retrieved from: https://www.3dtotal.com/interview/24-interview-with-infinite-realities-by-richard-tilbury-3d-lee-perry-smith-body-scanning

Tikka, P., Väljamäe, A., de Borst, A., Pugliese, R., Ravaja, N., Kaipainen, M., & Takala, T. (2012). Enactive cinema paves way for understanding complex real-time social interaction in neuroimaging experiments. Frontiers in human neuroscience, 6, 298.

Tikka, P., Vuori, R., & Kaipainen, M. (2006). Narrative logic of enactive cinema: Obsession. Digital Creativity, 17(4), 205-212.

Tricart, C. (2017). Virtual Reality Filmmaking: Techniques & Best Practices for VR Filmmakers. Taylor & Francis.

Triplegangers. 2017. Retrieved from http://www.triplegangers.com

Vitazko, M. (2017). Representing Humans in Mixed Reality. Retrieved from: https://medium.com/microsoft-design/representing-humans-in-mixed-reality-dc241ab97434

Warne, M. (2015). Photogrammetric software as an alternative to 3D laser scanning in an amateur environment.

Williams, L. (1990, September). Performance-driven facial animation. In ACM SIGGRAPH Computer Graphics (Vol. 24, No. 4, pp. 235-242). ACM.

Xia, J., Garcia, I., He, Y., Xin, S. Q., & Patow, G. (2011, February). Editable polycube map for GPU-based subdivision surfaces. In Symposium on interactive 3D graphics and games (pp. 151-158). ACM.Praesent blandit, cursus ornare. *Morbi porta (s. 53-56)*, orci ac auctor mollis, risus nisl tincidunt lacus, eu accumsan magna neque id nibh (2000).

Zollhöfer, M., Thies, J., Garrido, P., Bradley, D., Beeler, T., Pérez, P., ... & Theobalt, C. (2018, May). State of the Art on Monocular 3D Face Reconstruction, Tracking, and Applications. In Computer Graphics Forum (Vol. 37, No. 2, pp. 523-550).

7. List of Figures