TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Roman Hrushchak 166802IVSM

VISUALIZATION OF TONGUE AND LIP MOVEMENTS

Master's thesis

Supervisor: Einar Meister

Senior researcher,
PhD

Tallinn 2018

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Roman Hrushchak 166802IVSM

KEELE JA HUULTE LIIKUMISE VISUALISEERIMINE

Magistritöö

Juhendaja: Einar Meister

Vanemteadur,
PhD

Tallinn 2018

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Roman Hrushchak

05.05.2018

Abstract

The goal of the thesis was to create an application to visualize the tongue and lips movements by using the electromagnetic articulography (EMA) technology for a motion capture.

In result, 3D models of the tongue and lips with mapped virtual bones were created by using a box modelling technique and integrated into an existing virtual head model. To achieve a realistic animation of the tongue and lips movements, real-world data from the sensors attached to a human's face and tongue were collected using the EMA method. The NDI Wave Speech Research System was used to capture the real articulatory data, Blender for modelling purposes and Unity game engine was used to create a demo application.

The virtual head model is animated by applying kinematics to the virtual bones of the tongue and lips using the data provided by EMA sensors. The technical realization was accomplished by a direct mapping of sensor's position and spin to a virtual bone linked to a 3D model's mesh. In addition, the graphical user interface of the demo application provides panels for manual control of the head, the jaw, the eyebrows, and the tongue.

The source code scripts were written in C# script language using Microsoft Visual Studio environment.

This thesis is written in English and is 30 pages long, including 5 chapters, 23 figures and 2 tables.

Annotatsioon

Töö eesmärgiks oli luua keele ja huulte liikumise visualiseerimise rakendus, mis kasutab elektromagneetilise artikulograafia (EMA) abil saadud lähteandmeid.

Töö tulemusena loodi keele ja huulte 3D mudelid ja integreeriti need olemasoleva virtuaalse pea mudeliga. Keele ja huulte artikulatoorsete liigutuste realistlikuks animeerimiseks salvestati kõneleja keelele ja näole kinnitatud EMA sensorite liikumisandmeid. Artikulatsiooniandmete kogumiseks kasutati firma NDI EMA-süsteemi Wave Research System, keele ja huulte mudelid loodi Blenderis ning demorakendus loodi Unity mängumootorit kasutades.

Virtuaalse peamudeli animeerimisel juhitakse keele ja huulte virtuaalsete luude liikumist vastavalt EMA sensoritelt kogutud andmetele. Tehniliselt on see realiseeritud nii, et sensorite positsiooni- ja pöördenurga andmed edastatakse vastavatele 3D mudeli tippudele. Lisaks on demorakenduses võimalik käsitsi muuta pea, alalõua, kulmude ja keele asendit.

Rakenduse lähtekood on kirjutatud C# keeles Microsoft Visual Studio arenduskeskkonnas.

Magistritöö on kirjutatud inglise keeles ning sisaldab teksti 30 leheküljel, 5 peatükki, 23 joonist, 2 tabelit.

List of abbreviations and terms

TTU	Tallinn University of Technology
EMA	Electromagnetic articulography
MRI	Magnetic resonance imaging
3D	Three-dimensional
AV	Audio-visual
GUI	Graphic User Interface
NDI	Northern Digital Inc.
NURBS	Non-uniform rational B-Splines
VG	Vertex Groups
DOF	Degree-of-Freedom
IK	Inverse Kinematics
B-bone	Bezier bone
AV	Audio-Visual
RMS	Root mean square
UE	Unreal Engine

Table of contents

List of figures

List of tables

1 Introduction

Electromagnetic articulography (EMA) enables to capture 3D data of articulatory movements of the lips and tongue. The visualization of the captured data using 3D models of the lips and tongue could be used in computer-aided pronunciation training and in audio-visual speech synthesis. The thesis involves the development of 3D models of the lips and tongue controlled by the 3D movement data stored from the sensors of the EMA system.

1.1 Problem statement

Humans employ speech reading commonly in adverse listening conditions and in general to facilitate speech perception [1]. The ability to visually obtain phonetic information depends on seeing facial movements that are produced by the speech articulators: mainly by the lips and the jaw and to some extent by the larynx and the tongue. These movements have been shown to be highly correlated with speech acoustics [2]. They are also commonly called the articulatory gestures. The conversation in any natural language consist of the combination of several aspects of the language perception and, besides the audio information, they also include the visual information obtained from speech articulators.

During a coarticulation, an isolated speech sound is influenced by a preceding or succeeding sound. Thus, obtaining just the acoustic data is not enough for a proper evaluation of the human speech acquisition. The virtual talking heads are used in multimodal interactive systems allowing to perform speech visualization synchronously with the produced sounds. The main challenge of such virtual agents is the realistic visualization of a speech in the selected natural language.

The goal of this research is to create three-dimension tongue and lips models with virtual bones (controllers) attached to them. The realistic visualization of the tongue and lips movements during a speech should be done by using the EMA technology for a motion capture. The created three-dimension tongue will be animated by applying kinematics to the virtual bones using the fifth-dimension sensors to capture the movements of lips, tongue and chin as described in electromagnetic articulography (EMA) method [3]. The NDI Wave Speech Research System [4] will be used to capture data from the sensors.

Synchronized complex transitions between the articulatory gestures allow to distinguish phonemes which pronunciation depends mainly on the chosen natural language. In this research, the Estonian language has been chosen for the audio-visual recordings.

The alternative solutions will be examined and compared to the proposed approach. Quantitative research methodology will be used in this work. The results will be validated by comparing the animation of the created three-dimension virtual head with the actual real speech articulators' movements.

1.2 Organization of the Thesis

This thesis includes the next chapters:

1. **Background.** Contains the background and literature review. Existing solutions are analysed and compared.

2. **Tools and methods.** Brief overview of the existing modelling techniques is presented and main tools for 3D modelling and scripting are described.

3. **Application.** This chapter explains methods and tools used to create a talking head application for a demonstration purpose. It includes the program installation and usage overview.

4. **Summary.** Overview of the accomplished tasks and suggestions regarding the future work.

2 Background

Simulation of the tongue has important applications in biomechanics, medical science, linguistics, and graphics [5]. Learners of a foreign language, for instance, would get articulatory feedback to discover the articulation of sounds that do not exist in their mother tongue [6].

Several articulatory models of the tongue have been proposed over the years, approaching the complexity of the tongue muscles from different viewpoints and making different simplifications to arrive at a working model, depending on the relevant application. The main distinctions are between physiological and geometrical or statistical modelling, between two- (2D) and three-dimensional (3D) models, and if the model is real-time or not [7].

2.1 Literature review

Modern imaging techniques such as magnetic resonance imaging (MRI) have allowed the creation of highly detailed and anatomically accurate articulator models. However, due to the slow imaging acquisition rate of MRI (if high resolution is to be obtained), such models are constrained to static configurations of the articulators. Articulatory dynamics can be measured by techniques such as EMA which track the 3D movements of discrete points during speech [8].

Tongue movements are dynamic, and are modelled as such in biomechanical models, where an equation of motion is solved to determine the displacement of the model as a function of muscle activation and the effects of inertia. A simpler, and computationally faster, alternative is to focus on a kinematic description of the tongue, dealing with the trajectory of tongue movement, instead [7].

The paper [9] describes the coarticulation as a tongue shape for each phonetic entry used in speech which is influenced by the preceding and following phoneme. Thus, for a single phoneme there is more than one universal tongue shape and the coarticulation makes it

possible to combine vowels and consonants to obtain dozens of unique tongue forms for the same sound pronunciation. Consonants and vowels show distinctive characteristics; for vowels, the entire tongue surface matters as well as its curvature while for consonants it is mainly the areas of contact between the tongue and the palate or teeth that matter. For consonants, the tongue touches the palate with more tension than for vowel [6]. Here comes the need in a more detailed analysis of how a one phoneme implies the changes in the tongue shape of another.

In a related work [10], the EMA is described as a method to capture the location and orientation of the sensors, attached to the articulators, during speech. While the lip movements can be acquired using optical tracking and the teeth and jaw are rigid bodies, the tongue is more complex to model, since its anatomical structure makes it highly flexible and deformable [10].

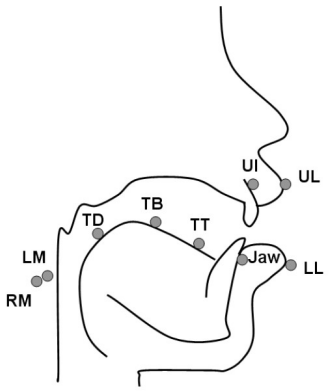

Figure 1. Schematic view of the speaker showing the position of the EMA sensors [2]

The authors of another research used an EMA system to record the position and orientation of sensors attached to the gum, lips and tongue. In the experiment, the position and orientation of nine sensors were recorded at 100 Hz. Two sensors were attached to the lips: on the upper lip and on the lower lip. Three sensors were glued on the tongue: TT on the tongue tip, TB on the tongue body and TD on the tongue dorsum. Two sensors (LM and LR) were also attached to the left and right mastoid processes to correct for head movement [2] (Figure 1). This 3D model was originally controlled by a set of animation frames. The visible and partially occluded speech facial articulators such as lips, jaw, and tongue. Linear interpolation was used to create the animation. Unfortunately, it was

impossible to accurately replicate speech articulator movements and authors had to develop a new animation method.

Selected key frames (from the original model) were used to create articulatory parameters for driving the avatar. Each tongue sensor from EMA data was associated with a specific vertex of the 3D tongue mesh of the original face model. For each sample of the quantized EMA database, tongue postures were determined by estimating the best linear mixture. Consequently, the method to derive the tongue model presented in this paper could be used to create a more accurate HD facial articulatory model [2].

The Wave Speech Research System (NDI) is an electromagnetic articulography (EMA) system that supports three-dimensional tracking of 5 or 6 degree-of-freedom (5-DOF, 6-DOF) sensors in one of two electromagnetic field volumes. 5-DOF sensors allow tracking of x, y, and z spatial coordinates, as well as angular coordinates characterizing rotation about the transverse axis (pitch) and anterior–posterior axis (roll). 6-DOF sensors have the added capacity for tracking angular coordinates characterizing rotation about the inferior–superior axis (yaw) [4]. The electromagnetic field generator itself is a mountable box which is placed in profile to the subject. The sensors' signals have a variable strength based on the distance and relative orientation between the receiver and the transmitter. Thus, sensor location can be derived from signal strength, but deviations in sensor orientation decrease the overall quality of measurements.

For articulated body animation, the 3D points are normally used as transformation targets for the rigid bones of a hierarchically structured skeleton model. The skeletal transformations are applied to a virtual character by deforming its geometric mesh accordingly [11]. The human tongue does not contain any bones and is extremely deformable. Due to this, it is beneficial to work with 3D vectors instead of just points. This is possible to achieve because each of the sensors tracks orientation in addition to a location.

To use the tongue motion capture data to control a tongue model using skeletal animation, we design a simple skeleton as a rig for the tongue mesh. This rig consists of a central spine, and two branches to allow lateral movement, such as grooving. The rig can be animated by using inverse kinematics (IK) to determine the location, rotation, and deformation of each Bezier bones (B-bone) for any given frame [11].

2.2 Existing solutions

2.2.1 EMA as motion capture

The article [12] introduces a data driven three-dimensional talking head system, in which the EMA-recorded data is used to synthesize articulatory movements, and then control the 3D articulator dynamics. This approach included pre-defined animations for the 3D head models used by the co-articulator blending and smoothing algorithm to generate the movement contour. The evaluation of the system has showed that the realism scores of 3D articulatory animations for minimal pairs were far from ideal (3.5 points in average, out of 5) and the overall identification accuracy was 91.6% among 286 tests [12]. This study did not analyse the effect of the co-articulation for longer sounds where the realism score could be even lower than with minimal pairs of words.

In paper [10] the authors describe a technique for articulatory animation by adapting a conventional motion capture based animation paradigm. The static 3D model was extracted from volumetric MRI and dental scans from the same speaker as the EMA data. The developed system relied exclusively on the articulatory data and used the spline IK based animation. Even though the overall mean correlation was 0.95, there weren't any constraints to prohibit the 3D models from passing through one another [10]. The future work of the reviewed system included the possibility to use longer co-articulation pairs preceding or ascending one another.

A simple dynamic tongue model was built and described in [13]. The authors decided not to use a realistic tongue with high precision in articulation because it would require a lot of efforts and the basic model was sufficient. The article demonstrates a possibility to use an articulatory data captured from a real human speaker in an analogous manner.

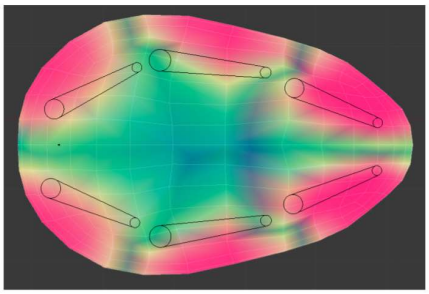

Figure 2. Top view of armature (six components) and an automatically assigned vertex weight map, with the colours indicating the degree of influence of each component on the vertices; the tongue mesh is visible as a wireframe (tip on right) [13].

Figure 2 demonstrates an example of a tongue armature and its automatic vertex weight map. Attaching many sensors to the tongue can be a challenge. The first task is to find the best placement map in such a way that the number of attached coils would be sufficient for the data acquisition and not too many wires should be involved. Next, there may be a problem of interference if the distance between the sensors is too short.

Ideally, for coils arranged along the mid-sagittal contour of the tongue, the coil axes should lie in the mid-sagittal plane, while for the lateral coils, a transverse axis orientation tends to provide a more robust indication of tongue grooving [13].

The described research work demonstrated a simple tongue model with working armature. Though, the authors did not integrate it in the audio-visual (AV) synthesizer nor they combined the tongue and facial animation together.

Figure 3. Side view of tongue mesh and armature [13]

Figure 3 shows a side view of the tongue mesh and armature. The centre and right panels show a bunched and retroflex configuration of the tongue model, respectively [13]. Also,

another problem is handling the noisy and missing data from the sensors which has not been solved in the reviewed paper. Moreover, a larger and phonetically balanced EMA corpus should be recorded and processed for a proper evaluation of the created tongue model.

2.2.2 A mass-spring tongue model with efficient collision detection

Combining the tongue animation with other speech related techniques can improve the effect of spoken language learning. In [14], a mass spring based real time tongue animation system with the functions of collision detection and response to stimulus was presented. Authors emphasize the importance of the collision detection for the realistic facial and tongue animation. The main problem of this research was that most of the existing solutions had satisfying 3D models for the animation but they had a lack of the response handling and collision detection, which substantively reduce the level of talking heads perception.

For an accurate and realistic visual speech synthesis, collision detection should be considered into simulated tongue. The depth of penetration of points was used to calculate the extrusion force generated from the collision between corresponding tongue surface points and inner face of teeth or upper jaw. After that, it leads to the local deformation of the regions of the tongue which have contacted with other articulators [14].

2.2.3 Ultrasound technology for a motion capture

The paper [15] presents a system that is developed for automatic extraction and tracking of the tongue surface movements from ultrasound image sequences. This technology is non-invasive and significantly less expensive than other technologies. Also, it can provide real-time capture rates (30 frames per second). Some of the main problems with the ultrasound imaging are artefacts, refractions, unrelated reflections and a corrupting noise. Moreover, the tongue surface can reflect ultrasound waves only in a range of angles thus making the motion analysis difficult [15]. In this paper, the tongue movements were tracked by extracting the contours from the captured images (two-dimension tracking).

The ultrasound tongue imaging tends to require extensive processing to track the mid-sagittal tongue contour and does not usually capture the tongue tip, while real-time magnetic resonance imaging (MRI) has a very low temporal resolution, and is currently possible only in a single slice [11].

3 Tools and methods

Modelling is simply the art and science of creating a surface that either mimics the shape of a real-world object or expresses your imagination of abstract objects [16]. This research requires a 3D model of a human head which should also contain a model of a tongue. Both models should be rigged (properly attached to respective bones) before the animation is applied. Data acquisition technics need to be used to control the virtual bones in a human-like way.

Finally, the demo application should consist of a simple graphic user interface and provide full control over the talking head model and its parts (jaw, lips, a tongue). This section describes different data acquisition methods and modelling tools.

3.1 Data acquisition methods and tools

To achieve a realistic animation of a talking head, when pronouncing complex joint phonemes, there was a need to use real-world data from the sensors attached to a human's face and tongue.

NDI Wave is the most accurate electromagnetic articulography speech research system on the market[1]. It allows to track articulatory orofacial movements in real-time and provides the precision enough to make a virtual agent behave in a realistic manner.

The Wave electromagnetic articulography speech research system measures the 3D position of micro sensors attached to the tongue, palate, lips and face for tracking orofacial movements. The resulting measurement data can be applied in the research of apraxia of speech, dysarthria, oral myo-functional disorders, and other speech-related pathologies. The Wave can simultaneously track up to 16 disposable sensors in five degrees of freedom (5DOF); up to 8 sensors in 6DOF. Measurements are captured in real time without requiring a line of sight between the system and subject. Two subjects can be tracked at once, allowing for the study of dyadic communication, with the results

[1] https://www.ndigital.com/msci/products/wave-speech-research/

displayed side-by-side in the software interface. High temporal accuracy and spatial resolution ensure the fastest and most subtle of movements are recorded [17].

Primary features [17]:

5. Minimal lag and noise result in exceptionally accurate data [4];
6. No line of sight requirements ensures uninterrupted 5DOF/6DOF tracking;
7. No post-processing of kinematic data reduces experiment time and errors;
8. Disposable micro sensors allow for natural subject movement;
9. Real-time synchronization of audio and kinematic data.

As an additional option, the Wave Speech Research System can be spatially and temporally synchronized with the Optotrak Certus[1] and 3D Investigator[2] motion capture systems to enable greater power and flexibility within research applications. Integrating a Northern Digital Inc. motion capture system offers the following benefits [17]:

1. Measure up to 512 data points;
2. Integrate and synchronize with additional research equipment.

Real-Time Application Development Software is also available, which includes an integrated TCP/IP server for real-time data streaming to third-party clients, and gives users the ability to create start/stop trigger commands from third-party clients [17].

Technical specification of NDI Wave is described in Table 1.

Table 1. Technical Specifications of NDI Wave

System Performance	
Static positional accuracy	0.6 mm RMS
Static angular accuracy	0.2 degrees RMS

[1] https://www.ndigital.com/msci/products/optotrak-certus/

[2] https://www.ndigital.com/msci/products/3d-investigator/

Dynamic positional accuracy	1.5 mm RMS
Dynamic angular accuracy	0.6 degrees RMS
Number of sensors tracked simultaneously	8 – using 8 channel system; 16 – with extension package
Sample Frequency	100 Hz (Standard) 200 Hz, 400 Hz (Optimal)
Field Generator	
Dimensions (H x W x D)	200 mm x 200 mm x 80 mm
Weight	3.2 kg
System Control Unit	
Dimensions (H x W x D)	88 mm x 235 mm x 295 mm
Weight	3.4 kg
Sensor Interface Unit	
Dimensions (H x W x D)	32 mm x 50 mm x 90 mm
Weight	250 g

3.2 Modelling techniques

Box modelling is a polygonal modelling technique in which the artist starts with a geometric primitive (cube, sphere, cylinder, etc.) and then refines its shape until the desired appearance is achieved. Box modelers often work in stages, starting with a low-resolution mesh, refining the shape, and then sub-dividing the mesh to smooth out hard edges and add detail. The process of subdividing and refining is repeated until the mesh contains enough polygonal detail to properly convey the intended concept. Box modelling is probably the most generic form of polygonal modelling and is often used in conjunction with edge modelling techniques [18].

Edge modelling is another polygonal technique, though fundamentally different from its box modelling counterpart. In edge modelling, rather than starting with a primitive shape and refining, the model is essentially built piece by piece by placing loops of polygonal

faces along prominent contours, and then filling any gaps between them [18]. This sculpting method can be used to create a tongue model to use in the demo application.

NURBS is a modelling technique used most heavily for automotive and industrial modelling. In contrast to polygonal geometry, a NURBS mesh has no faces, edges, or vertices. Instead, NURBS models are comprised of smoothly interpreted surfaces, created by "lofting" a mesh between two or more Bezier curves (also known as splines) [18].

Image based modelling is done by obtaining 3D objects from a set of static two-dimensional images using sophisticated algorithms.

3D Scanning is a method of digitizing real-world objects when an incredibly prominent level of photo-realism is required. A real-world object (or even actor) is scanned, analysed, and the raw data (typically an x,y,z point cloud) is used to generate an accurate polygonal or NURBS mesh [18]. This is a great method to obtain a photo realistic and close to human-like models of a jaw, a tongue and teeth.

3.3 Modelling tools

There are dozens of software products which help to create professional high-quality 3D models. The key features needed to develop a talking head are mesh constructing and modification, armature creation, paint weighting, support for blend-shapes and exporting the scene to a separate application (in this case — game engine).

The most popular 3D modelling software products are determined by customer satisfaction and scale:

1. Cinema 4D;
2. 3ds Max Design;
3. Blender;
4. Maya;
5. ZBrush;
6. LightWave 3D.

Blender is a free and open source 3D creation suite, it supports the entirety of the 3D pipeline — modelling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation[1].

Blender is supported by community and has enough guidelines and tutorials. The documentation (reference manual[2]) covers all aspects of the 3D pipeline and is sufficient to develop the required models of a human head and its parts. Also, it is cross-platform and its source code can be compiled to other operation systems in the future.

Other 3D modelling suites propose a wider selection of features and tools but also require much more time to learn the interface and their complexity is excessive. Compared to them, Blender also has a well-established integration with Unit game engine which saves time and provides a slightly more predictable behaviour.

3.4 Game engine

Demo application should be user-friendly and offer the ability to control the parameters of head, jaw and tongue models. The easiest way to achieve this is to use a game engine – a special software meant to develop games or any 3D applications with complex animations, allowing to apply textures and materials and changing objects' parameters in the process.

Usually, a game engine is a set of components, tools, graphical user interface (GUI) elements, scripting engine, which offers, at least, a basic functionality:

1. Working with 3D objects (and render them as graphics);
2. Light system;
3. Processing sound;
4. Network components (to create more complex, interconnected applications).

[1] https://www.blender.org/

[2] https://docs.blender.org/manual/en/dev/

Game engines provide many assets: modules, libraries, effects, and tools – so there is no need to create everything from scratch.

The comparison of the most popular game engines of 2017 which are free to use for non-paid applications is shown in the Table 2.

Table 2. Game engines cross-platform overview 2017[1]

	Windows	Linux	Mac	Android	iOS	Playstation	Xbox
Unreal Engine (UE)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Unity	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CryEngine	Yes	Yes	No	Yes	Yes	Yes	No
Godot Engine	Yes	Yes	Yes	Yes	Yes	Yes	No
Amazon Lumberyard	Yes	No	No	Yes	Yes	Yes	Yes
ShiVa Engine	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Most of these software products require a lot of time to learn them. Game engines like UE and CryEngine allow to develop photo-realistic games and tools but they are over complicated for a project like talking head.

The final choice has been made and Unity has been selected as a great tool to make a working prototype in brief time. This engine allows to build applications for many operations systems: Windows, Mac, iOS, Android, Playstation, Xbox.

Also, the simplicity of its interface allows a developer to focus on fulfilling the product requirements rather than dealing with interface and complex scripting language.

[1] https://blog.instabug.com/2017/12/game-engines/

4 Application

The parametric approach has been used to solve the problem of a realistic facial and tongue animation of a virtual agent. The created model is a 3D model of a human head which also includes a 3D model of tongue. To achieve a human-like animation, the 3D mesh of the tongue has been divided into several vertex groups (VG).

Using the Blender's automatic bone weighting tool, the created bones were aligned to the model of a tongue and then vertex groups were generated. It was achieved by first selecting the mesh and armature and then applying a modifier called "Armature Deform" by using the combination of keys "Ctrl+P" and selecting the option "With Automatic Weights" (Figure 4).

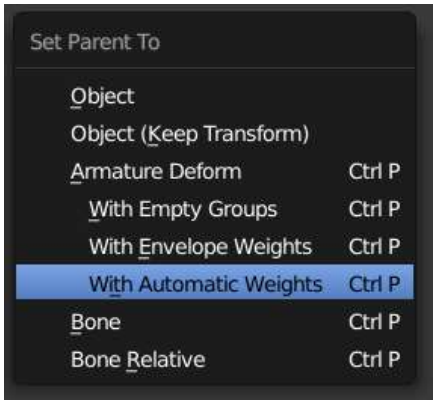

Figure 4. Blender's command to create Automatic Weights

This algorithm recalculates the bone weights based on the distance between the bones and the selected mesh. It allows to automatically set levels of influence for each existing bone on the mesh deformation. As the tool's algorithm is not perfect, the additional work with weight colouring was required to make deformations look more realistic. In case of the head model, no additional work has been needed because the used armature included all the necessary bones and weights.

4.1 Rigging

Armature in Blender represents an armature of a real world's skeleton and consists of many bones. It is also a type of object which is used for rigging. Objects, attached to bones, will move and deform in the same way and are like a real-world puppet. The

created rig then allows to animate objects or export them to external programs. The armature object has its origin and parameters as rotation, position and scale.

The used head model's armature, containing the assigned bones and defined vertex groups, is displayed on the Figure 5. The main model bones are connected to a neck, a head, a jaw and eyebrows. On the other hand, the tongue model has bones for the base part, back part, middle part and a tongue tip (Figure 6Figure 5).

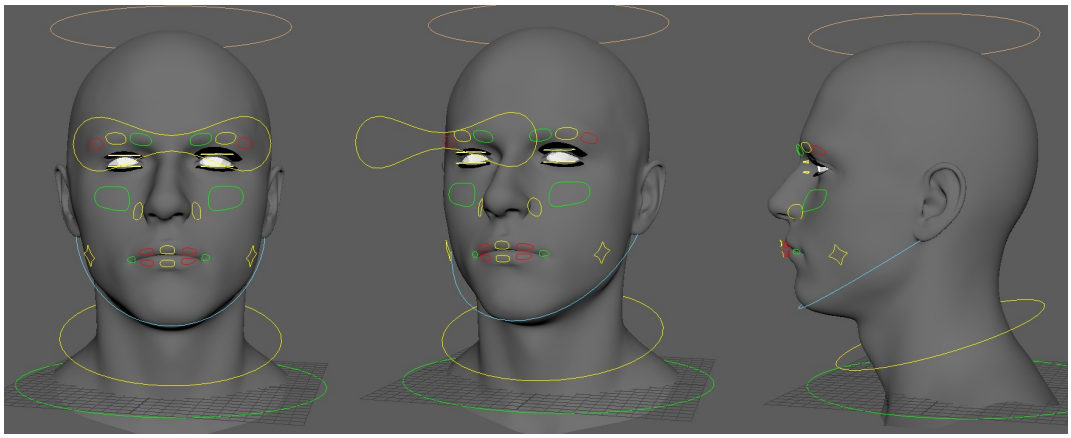

Figure 5. Head model and the assigned armature

Vertex groups can potentially have a very large number of associated vertices and thus many weights (one weight per assigned vertex). Weight painting is a method to maintain substantial amounts of weight information in a very intuitive way. It is primarily used for rigging meshes, where the vertex groups are used to define the relative bone influences on the mesh [19]. Blender provides a user-friendly and intuitive way to automatically assign weights to other objects (mesh, particles). In this work, the armature for the tongue model was created and vertex groups were automatically linked to the 3D model as shown on the Figure 7.

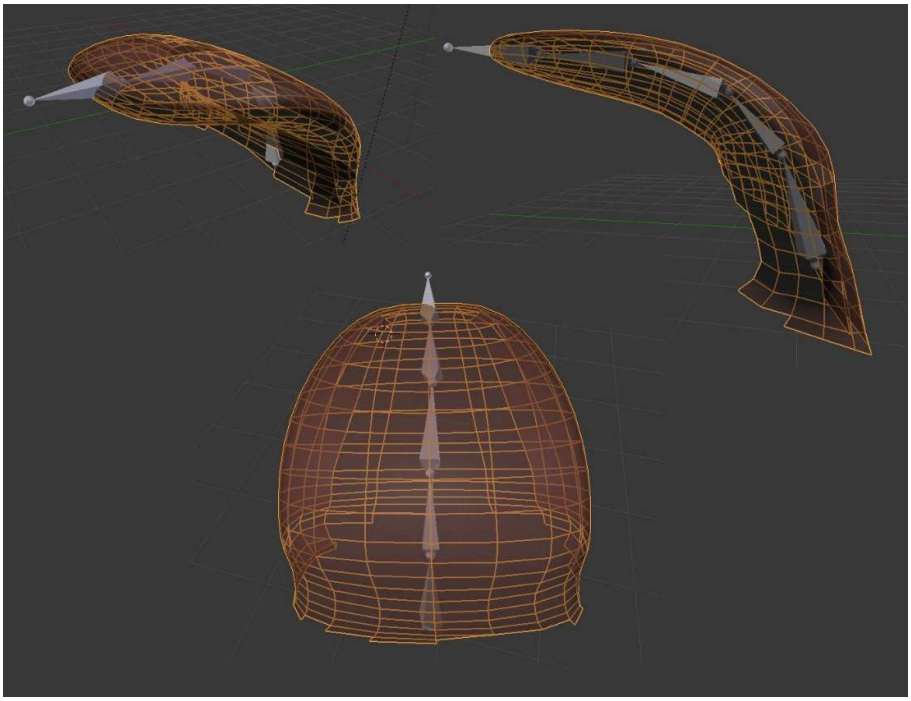

Figure 6. Tongue model and connected bones

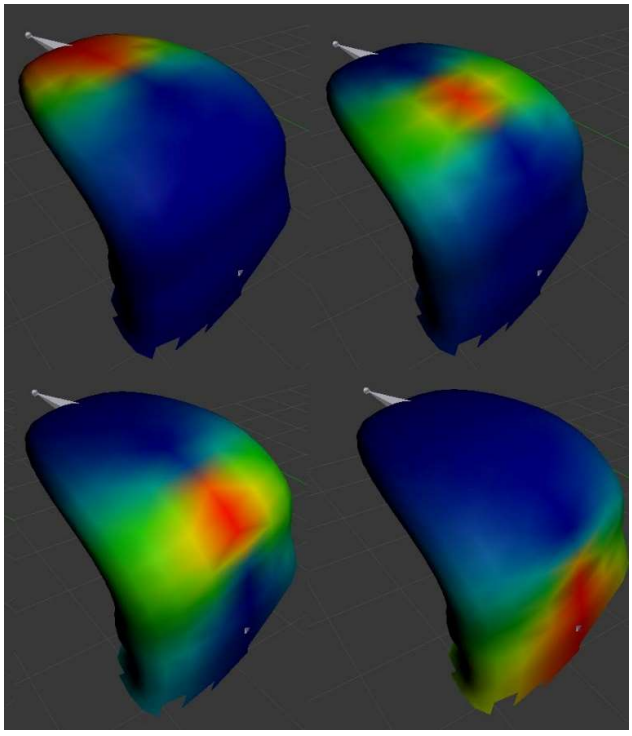

Figure 7. Weight painted tongue model

The figure above shows how different vertex groups (tongue sections) influence the deformation of the tongue mesh. From left to right, top to bottom: tongue tip, tongue middle part, tongue back part, tongue base. It is important that each of the model's parts would also slightly deform vertices of the connected part to make the final animation look more smooth and natural. The list of created vertex groups is shown on the Figure 8.

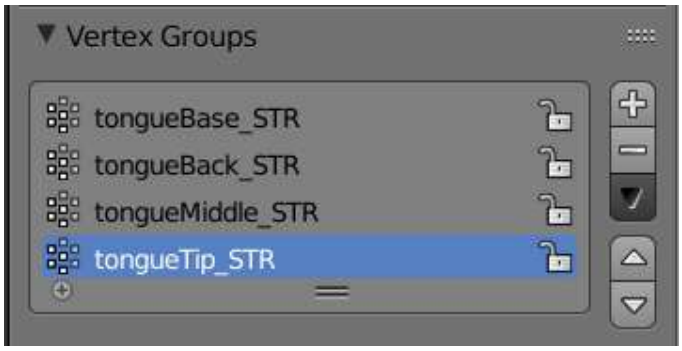

Figure 8. Vertex groups of the tongue 3D model

Weights are visualized by a gradient using a cold/hot color system, such that areas of low value (with weights close to 0.0) are drawn in blue (cold) and areas of high value (with weights close to 1.0) are drawn in red (hot). And all in-between values are drawn in rainbow colors (blue, green, yellow, orange, red) [19]. The weight color spectrum is shown on the Figure 9.

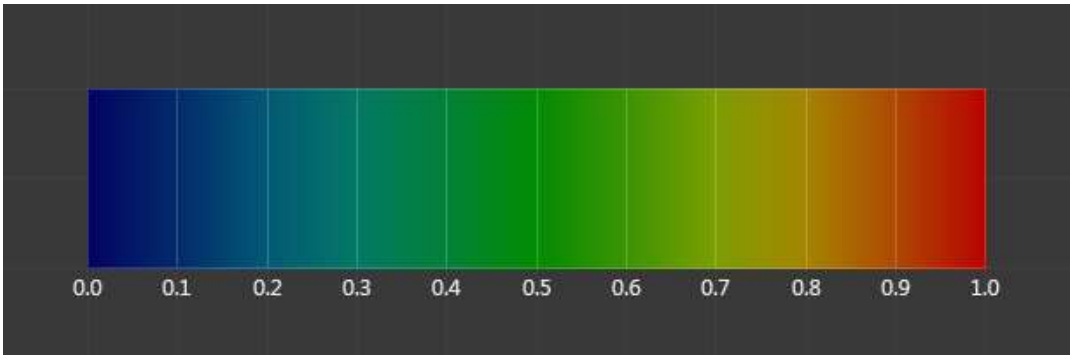

Figure 9. The colour spectrum and the respective weights [19]

In general, the part of the mesh around a bone should be colored in red and the farther vertices are from it, the colder color should be applied to them (a fade to blue gradient).

4.2 Animation

This research is focused on a parametric model and two options were analysed to animate the talking head. First approach is to use so called shape keys (SK) which allows to morph a model between two shapes based on a value from 0 to 1 (Figure 10).

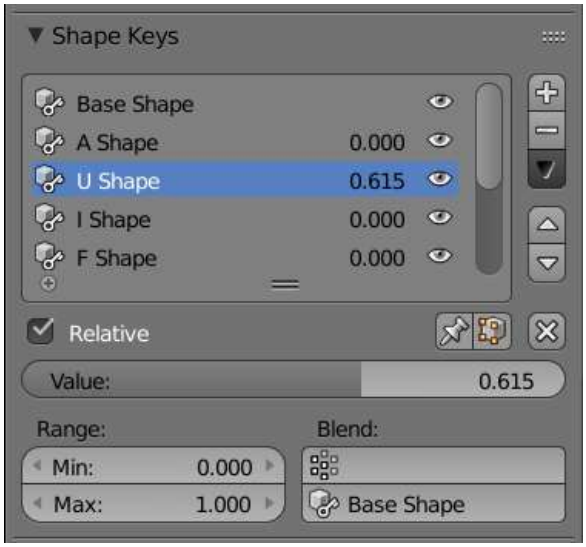

Figure 10. Example of configuring shape keys

Benefits of this method are simplicity of controls and smoothness of the resulting animation. Though, the main downside of it is the lack of realism. The real human face contains hundreds of muscles and, in this manner, is not limited in actions like shape keys which are used in the application.

The second approach is a direct mapping of sensor's position and spin to a virtual bone linked to a 3D model's mesh. The obvious advantage of this process is that the model will behave the same as the sensor attached to a real human face or a tongue. The weakness is the incapability to guarantee an evenness of the animation. To compensate the mentioned above problem, external tools can be used for filtering and smoothing the input data from outliers (R, VisArtico).

Considering the difference of both approaches, the second option has been selected as a preferred way to animate the talking head. Thus, there is no need to bake the animation or create shape keys beforehand – it will be done in real-time using the game engine capabilities.

The 3D models of a head and a tongue should be exported for using in an external game engine. Exporting is done by executing a command: File->Export->FBX (.fbx) as shown on the Figure 11.

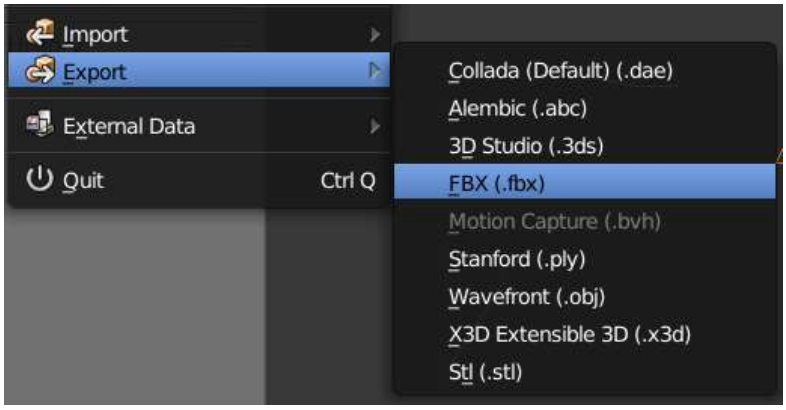

Figure 11. Blender's dialog for exporting a 3D model

To include blend-shapes (shape-keys) in the exported model, the option "Apply Modifiers" should be unchecked (Figure 12).

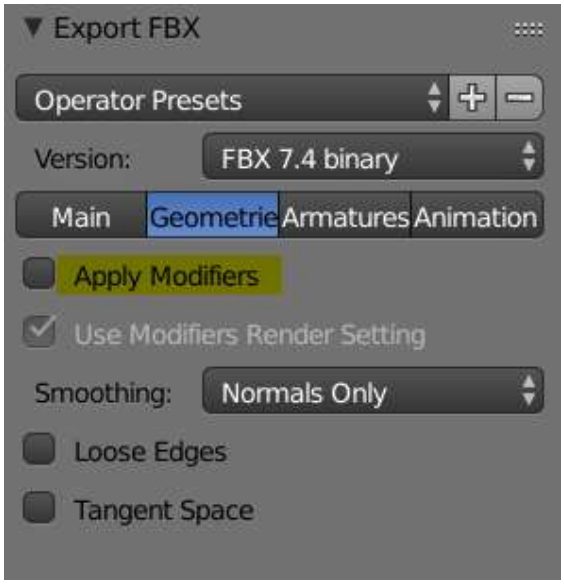

Figure 12. Export options (to support shape-keys)

The final model can be imported to Unity3D (or other respective game engine) and then transformed by changing the value for each blend-shape or by directly changing the position and rotation of bones attached to the model's armature (Figure 13).

The imported bones can be controlled directly by using the transform panel (Figure 14) of Unity3D or by modifying their values in a programmable way (using a script).

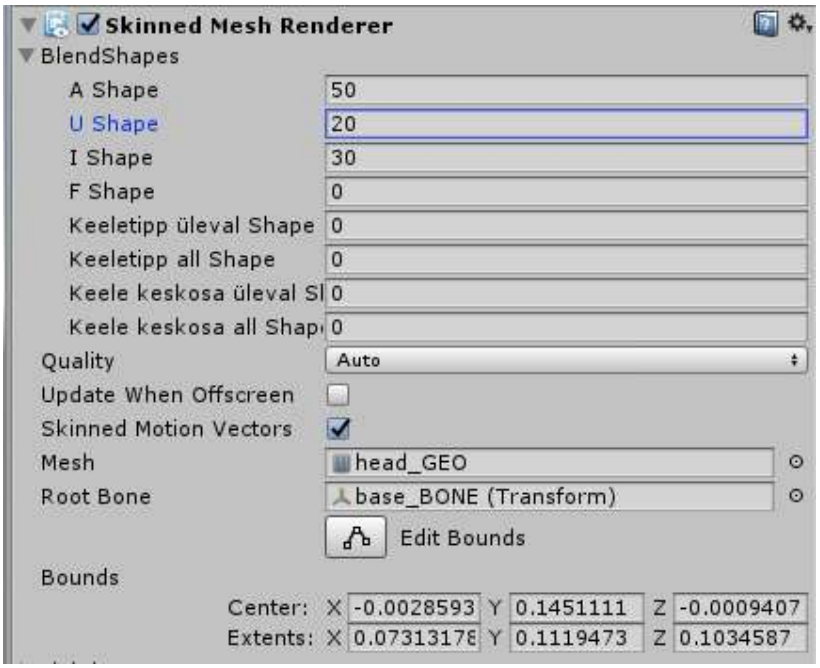

Figure 13. Blend-shapes control panel in Unity3D

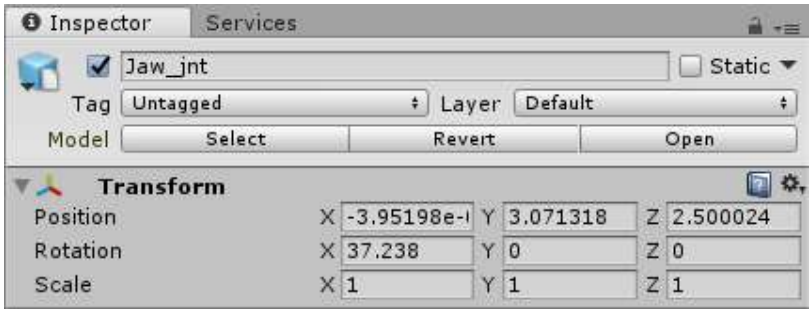

Figure 14. Unity3D object's transform panel

It should be mentioned that modifying the values of local transform will result in a transformation relative to the parent object, while the global transform changes the object's parameters related to the virtual world (zero coordinate). In this work, only the local transform of the respective objects (head, jaw, mouth's corners, tongue) was altered.

4.3 Installation

Unity3D provides a toolchain to build applications for many different platforms. The available export options include:

1. Windows (x32 and x64);
2. Linux;
3. Mac;
4. WebGL (HTML5 support is required);
5. Android;
6. iOS.

Demo project and the test-data (recordings) are stored in a zip-file on the attached media device. Furthermore, the source code of the application can be downloaded from the GitHub repository[1]. The project has been developed in Unity 2017.1.1f1 and Blender 2.79 was used to export the models and respective connected armatures.

Since some operation system would not run non-signed applications, the next steps should be done to bypass the security:

1. Windows: Run as administrator and allow 3rd party applications;
2. Mac: Allow applications downloaded from anywhere through the Security & Privacy settings.

The developed application has been tested on Windows 10 (x64).

[1] https://github.com/R-Hrushchak/thesis2018_talking_head

4.4 Application usage

The GUI (graphical user interface) of the developed application is very simple and intuitive. The application starts with opening a dialog for the file selection of recorded data in TSV format (Figure 15).

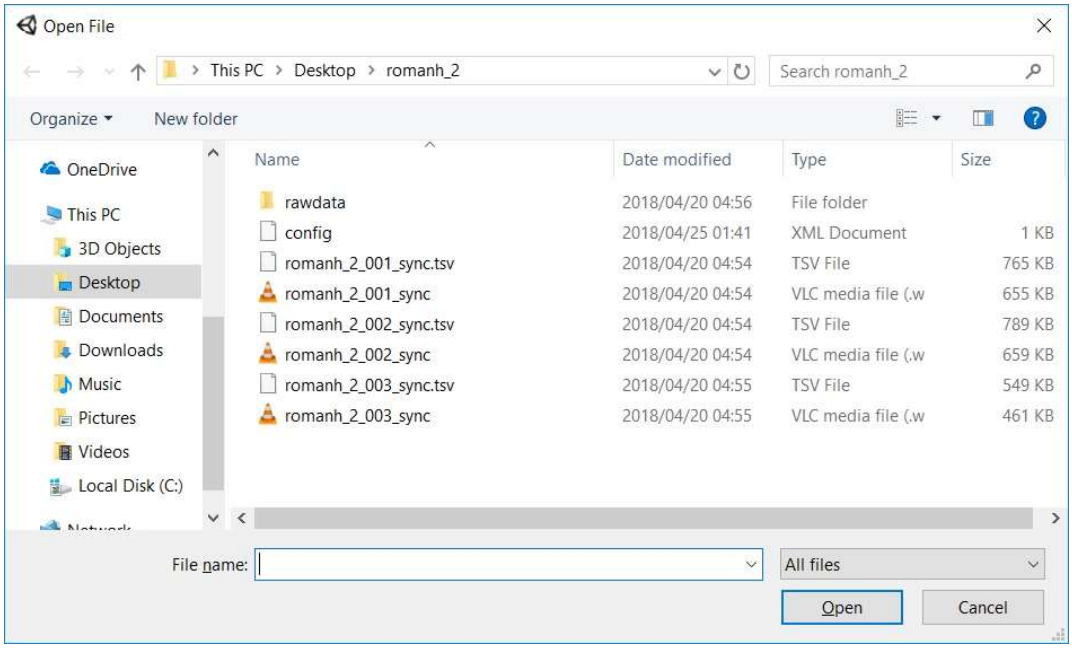

Figure 15. File dialog to select the recorded data file (TSV format)

The audio-track should be in WAV format and located in the same directory with the TSV-file to be properly loaded. After selection the recorded data file, the next step is to map the real sensors' numbers to the corresponding virtual bones used in the application (Figure 16). The inputs of the sensors configuration panel are marked as follows:

1. Reference sensor (usually, attached to a forehead);
2. Jaw sensor (attached to a jaw, under a bottom lip);
3. Lip corner left sensor (attached to the left intersection of lips);
4. Lip corner left sensor (attached to the right intersection of lips);
5. Tongue tip sensor (attached to the tip of a tongue);
6. Tongue mid sensor (attached to the middle part of a tongue);

7. Tongue back sensor (attached to the back part of a tongue);

8. Button “Close” (to close the sensor configuration panel);

9. Button “Quit” (to terminate the application).

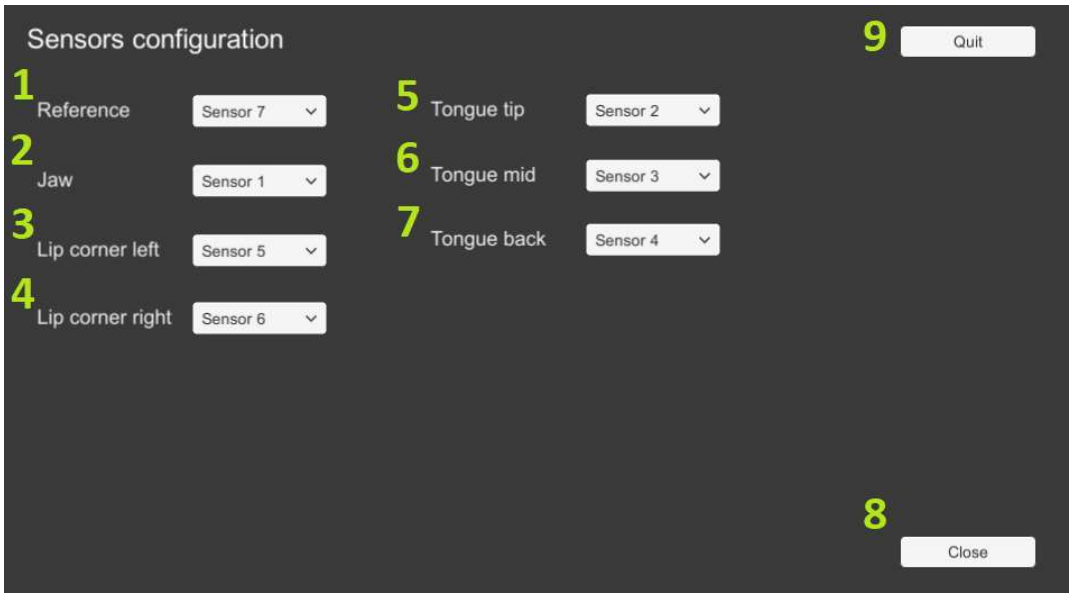

Figure 16. Sensors configuration panel

This panel can also be accessed (or closed) by the pressing the key “ESC”. After the sensors are mapped, the panel may be closed. The main application screen should appear as shown on the Figure 17.

Key elements of the main screen:

1. Manual control panel (provides the direct controls of the parametric model);

2. Button “Reset” (to stop the animation and reset the model);

3. Button “Play” (to start animation).

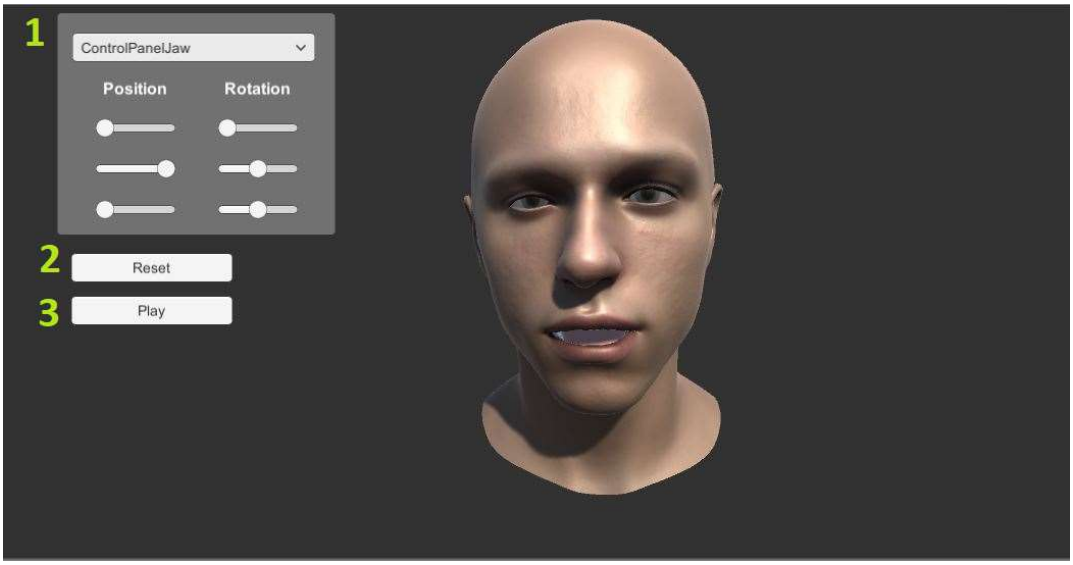

Figure 17. Main application screen

The jaw manual control panel (Figure 18) consists of the next elements:

1. Panel selector (head, jaw, eyebrows);
2. Position coordinate X;
3. Position coordinate Y;
4. Position coordinate Z;
5. Rotation value X;
6. Rotation value Y;
7. Rotation value Z.

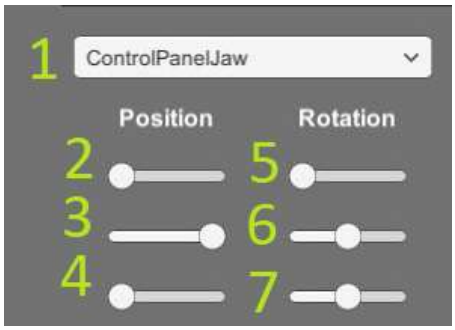

Figure 18. Jaw control panel

The head manual control panel (Figure 19Figure 18) consists of the next elements:

1. Panel selector (head, jaw, eyebrows);
2. Position coordinate X;
3. Position coordinate Y;
4. Position coordinate Z;
5. Rotation value X;
6. Rotation value Y;
7. Rotation value Z.

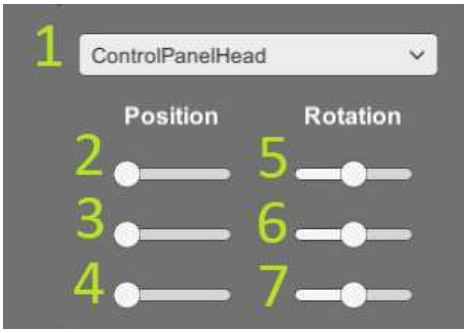

Figure 19. Head control panel

The eyebrows manual control panel (Figure 20Figure 19Figure 18) consists of the next elements:

1. Panel selector (head, jaw, eyebrows);
2. Position coordinate Y.

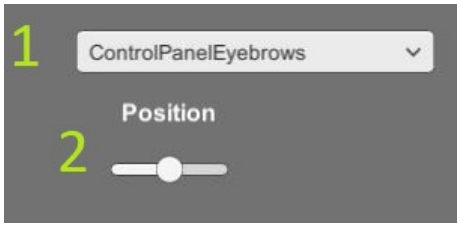

Figure 20. Eyebrows control panel

The tongue manual control panel (Figure 21Figure 20Figure 19Figure 18) consists of the next elements:

1. Panel selector (head, jaw, eyebrows);
2. Rotation value X (for the tongue's tip, middle and back);
3. Rotation value Y (for the tongue's tip, middle and back);
4. Rotation value Z (for the tongue's tip, middle and back).

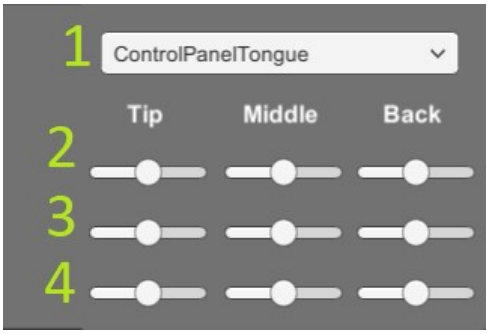

Figure 21. Tongue control panel

The sliders of the manual control menus will not cause any effect while playing the animation. Also, each bone, linked to a slider, has its minimum and maximum boundaries with allowed values. Animation playback is done from the script and has no limitations for spin and position, so having an incorrect or unfiltered data may produce an undesired output (glitchy animation). The reset button stops the animation, audio-track playback and sets the virtual talking head's model to its initial position and rotation. To play another animation file, the application should be restarted and the new data should be loaded again.

5 Summary

The main goal of this thesis was to create an application to visualize the tongue and lips movements by using the electromagnetic articulography technology for a motion capture.

Electromagnetic articulography enables to capture articulatory movements of a jaw, lips and a tongue. The visualization of the captured data using a talking head model could be used in a computer-aided pronunciation training and in audio-visual speech synthesis.

As a result, 3D models of the tongue and lips were created by using a box modelling technique, mapped to virtual bones and integrated into an existing virtual head model. The realistic animation of the tongue and lips movements was achieved by capturing a real-world data from the sensors, attached to a human's lips and tongue (Figure 22), using the EMA method.

Figure 22. EMA sensors attached to lips and a tongue

The NDI Wave Speech Research System was used to capture the real articulatory data. Blender was used for modelling purposes and the demo application was created using Unity game engine. The graphical user interface of the demo application provides panels for manual control of the head, the jaw, the eyebrows, and the tongue.

The side-by-side demonstration of a real person recording and the produced animation is shown on the Figure 23.

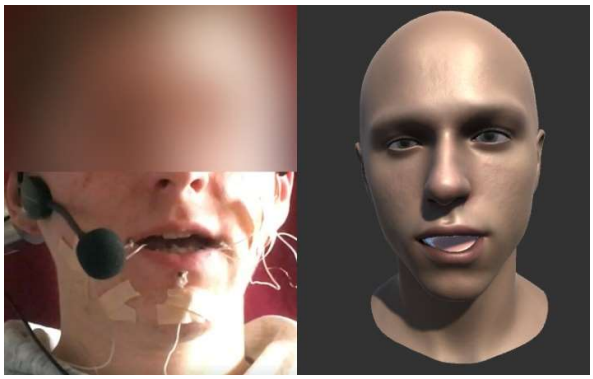

Figure 23. Side-by-side comparison of a real person and the talking head

The graphical user interface of the demo application provides panels for manual control of the head, the jaw, the eyebrows, and the tongue.

5.1 Future work

The created application may be improved by adding built-in filtering algorithms to remove outliers from the input dataset. This will produce much smoother animations without the need to use external applications to process the sensors data.

Also, the program interface can be extended with a file viewer which would allow to select a project directory and switch between the animation files (recorded data in TSV format).

The tongue model, used in the demo application, already produces realistic simulation but it may be also improved by adding more bones and using more sensors for the input data. For the better visualization, there could also be an option to control a transparency of the talking head model (for a better tongue view).

Finally, since the NDI software can stream sensors data through the network, this demo application could also have an option to accept such streams and synchronize the animation with a real person's actions with minimal delays. This will allow to create additional modules to help learning new languages or help people with disabilities to retrieve the ability to speak by providing almost a real-time feedback.

References

[1] G. Gibert, K. N. Olsen, Y. Leung and C. J. Stevens, "Transforming an embodied conversational agent into an efficient talking head: from keyframe-based animation to multimodal concatenation synthesis," , 2015 2015. [Online]. Available: https://link.springer.com/article/10.1186/s40469-015-0007-8. [Accessed 13 11 2017].

[2] G. Gibert, V. Attina, M. Tiede, R. L. Bundgaard-Nielsen, C. Kroos, B. Kasisopa, E. Vatikiotis-Bateson and C. T. Best, "Multimodal speech animation from electromagnetic articulography data" , 2012 2012. [Online]. Available: http://eurasip.org/proceedings/eusipco/eusipco2012/conference/papers/1569581799.pdf. [Accessed 13 11 2017].

[3] A. K. Pattem, A. Illa, A. Afshan and P. K. Ghosh, "Optimal sensor placement in electromagnetic articulography recording for speech production study," *Computer Speech & Language,* vol. 47, no. , pp. 157-174, 2018 2018.

[4] J. J. Berry, "Accuracy of the NDI wave speech research system.," *Journal of Speech Language and Hearing Research,* vol. 54, no. 5, pp. 1295-1301, 2011 2011.

[5] J. Yu, C. Jiang and Z. Wang, "Creating and simulating a realistic physiological tongue model for speech production," *Multimedia Tools and Applications,* vol. 76, no. 13, pp. 14673-14689, 2017 2017.

[6] M. Aron, A. Toutios, M.-O. Berger, E. Kerrien, B. Wrobel-Dautcourt and Y. Laprie, "Registration of multimodal data for estimating the parameters of an articulatory model," , 2009 2009. [Online]. Available: http://dblp.uni-trier.de/db/conf/icassp/icassp2009.html. [Accessed 13 11 2017].

[7] O. Engwall, "Combining MRI, EMA and EPG measurements in a three-dimensional tongue model ☆," *Speech Communication,* vol. 41, no. , pp. 303-329, 2003 2003.

[8] X. B. Lu, W. Thorpe, K. Foster and P. Hunter, "From experiments to articulatory motion-A three dimensional talking head model," , 2009 2009. [Online]. Available: http://dblp.uni-trier.de/db/conf/interspeech/interspeech2009.html. [Accessed 13 11 2017].

[9] C. Pelachaud, C. W. A. M. v. Overveld and C. Seah, "Modeling and animating the human tongue during speech production," , 1994 1994. [Online]. Available: http://repository.upenn.edu/cgi/viewcontent.cgi?article=1065&context=hms. [Accessed 13 11 2017].

[10] I. Steiner, K. Richmond and S. Ouni, "Speech animation using electromagnetic articulography as motion capture data," *arXiv: Human-Computer Interaction,* vol. , no. , pp. 55-60, 2013 2013.

[11] I. Steiner and S. Ouni, "Progress in animation of an EMA-controlled tongue model for acoustic-visual speech synthesis," *arXiv: Artificial Intelligence,* vol. , no. , pp. 245-252, 2012 2012.

[12] L. Wang, H. Chen, S. Li and H. M. Meng, "Phoneme-level articulatory animation in pronunciation training," *Speech Communication,* vol. 54, no. 7, pp. 845-856, 2012 2012.

[13] I. a. S. O. Steiner, “Towards an articulatory tongue model using 3D EMA,” Université Nancy, Villers-lès-Nancy, France, 2012.

[14] R. Li, J. Yu, C. Jiang, C. Luo and Z. Wang, “A mass-spring tongue model with efficient collision detection and response during speech,” , 2014 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6936586. [Accessed 13 11 2017].

[15] Y. S. Akgul, C. Kambhamettu and M. Stone, “Automatic extraction and tracking of the tongue contours,” *IEEE Transactions on Medical Imaging,* vol. 18, no. 10, pp. 1035-1045, 1999 1999.

[16] “Introduction to modelling,” [Online]. Available: https://docs.blender.org/manual/en/dev/modeling/introduction.html.

[17] “NDI Wave Overview,” [Online]. Available: https://www.ndigital.com/msci/products/wave-speech-research/.

[18] “Common Modeling Techniques,” [Online]. Available: https://www.lifewire.com/common-modeling-techniques-for-film-1953.

[19] “Blender Weight Paint Introduction,” [Online]. Available: https://docs.blender.org/manual/en/dev/sculpt_paint/painting/weight_paint/introduction.html.

[20] K. M. Hiiemae and J. B. Palmer, “TONGUE MOVEMENTS IN FEEDING AND SPEECH,” *Critical Reviews in Oral Biology & Medicine,* vol. 14, no. 6, pp. 413-429, 2003 2003.

[21] “Best 3D Modeling Software,” 2018. [Online]. Available: https://www.g2crowd.com/categories/3d-modeling.