Project Title:	**i-Treasures**: Intangible Treasures – Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures
Contract No:	FP7-ICT-2011-9-600676
Instrument:	Large Scale Integrated Project (IP)
Thematic Priority:	ICT for access to cultural resources
Start of project:	1 February 2013
Duration:	48 months

Deliverable No: D5.5

Final Version of 3D Visualization for Sensorimotor Learning

Due date of deliverable:	30 November 2015
Actual submission date:	31 March 2016
Version:	10
Main Authors:	Selami Çiftçi (TT)

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 600676.

Project ref. number	ICT-600676
Project title	i-Treasures - Intangible Treasures – Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures

Deliverable title	Final Version of 3D Visualization for Sensorimotor Learning
Deliverable number	D5.5
Deliverable version	10
Previous version(s)	1-9
Contractual date of delivery	30 November 2015
Actual date of delivery	31 March 2015
Deliverable filename	D5_5_v10.docx
Nature of deliverable	P
Dissemination level	PU
Number of pages	56
Work package	5
Partner responsible	TT
Author(s)	Selami Çiftçi (TT), Nikos Grammalidis, Kosmas Dimitropoulos, Alexandros Kitsikidis, Filareti Tsalakanidou (CERTH), Francesca Dagnino, Francesca Pozzi(CNR), Clemence Leboullenger (UPMC), Patrick Chawah (CNRS), Sohaib Laraba, Stephane Dupont (UMONS), Sotiris Manitsaris (ARMINES), Christina Volioti (UOM) , Vasilis Charisis, Leontios Hadjileontiadis (AUTH)
Editor	Selami Çiftçi (TT)
EC Project Officer	Alina Senn

Abstract	This deliverable reports the final version of 3D Visualization for Sensorimotor Learning module. There are eight games representing the four different use cases of the i-Treasures Project. Besides, a new game framework was developed, which is designed to easily create any dance or body-motion-based game. The games are integrated with the i-Treasures platform and the LMS. However, there is some additional work to be completed, before the applications can be demonstrated to the public. Most of the challenging technical work has been completed and the remaining tasks are mostly related with polishing and increasing the usability.
Keywords	3D Visualization Module, Gamification, AI, Sensor Integration, Visualization, Virtual Reality, Sensorimotor Learning

Signatures

Written by	**Responsibility- Company**	**Date**
Selami Çiftçi	Responsible for D5.5 (TT)	21.03.2016
Verified by		
Selami Çiftçi	Task 5.2 Leader (TT)	25.03.2016
Athanasios Manitsaris	WP5 Leader (UOM)	25.03.2016
Approved by		
Nikos Grammalidis	Coordinator (CERTH)	31.03.2016
Kosmas Dimitropoulos	Quality Manager (CERTH)	31.03.2016

Table of Contents

List of Abbreviations

2D	Two Dimensional
3D	Three Dimensional
3DVMSL	Three Dimensional Visualization Module for Sensorimotor Learning
BGDCAM	Body and Gesture Data Capture and Analysis Module
ICH	Intangible Cultural Heritage
CMS	Content Management System
DTW	Dimensional Time Warping
EEG	Electroencephalography
EMA	Electromagnetic Articulography
FEM	Finite Elements Model
FIS	Fuzzy Inference System
GUI	Graphical User Interface
HBB	Human Beat Boxing
HMM	Hidden Markov Model
ICH	Intangible Cultural Heritage
IMI	Intangible Musical Instrument
LMS	Learning Management System
MRI	Magnetic Resonance Imaging
NPC	Non-player Character
SG	Serious Game
SIFT	Scale Invariant Feature Transform
TCP/IP	Transmission Control Protocol / Internet Protocol
UDP	User Datagram Protocol
US	Ultra Sound
XML	Extendible Markup Language

1. Executive Summary

Deliverable D5.5 presents the final version of 3D visualization module, which provides support for learning or mastering different types of selected intangible cultural heritage (ICH) use cases. In the context of the i-Treasures project four use cases have been selected: rare dance interactions, rare traditional songs, traditional craftsmanship and contemporary music composition. Game-like applications for sensorimotor learning were implemented for all the use cases based on educational game scenarios that are provided in "D5.2: First Version of 3D Visualization for Sensorimotor Learning". The games are designed to get input from different sensors or game devices, such as Microsoft Kinect v1 and v2, Leap Motion, Animazoo, the prototype hyper-helmet developed within the project and other types of sensors. They are implemented in Unity 3D game engine, which has support for all major mobile, VR, desktop, console, and TV platforms plus the Web.

The Final version of the 3D visualization module for sensorimotor learning supports eight ICH game-like applications:

1. Human Beatbox (HBB) game
2. Byzantine music game
3. Tsamiko game
4. Calus game
5. Walloon game
6. Contemporary music composition game
7. Virtual pottery game
8. Real pottery game.

In addition, a novel "generic framework for the creation of dance/body-motion-based games" was developed that allows experts to design and create their own dance games by just changing either built-in assets or in-house-captured motion data. We believe that this framework is an important contribution as it could be also used for creating educational game-like applications for many other domains related to human motion, such as physical exercise, physiotherapy, rehabilitation, etc.

Another important contribution in this Deliverable is the design and development of a low-cost singing game (for Byzantine music), which can be extended in the future to support additional use cases in the future. Specifically, the HBB game makes use of the expensive ultrasound device, a fact that severely limits the possibility to market this application for wide use. Thus, in the Byzantine music game, the idea is to use only very low-cost sensors, such as a Kinect device and a microphone (either the Kinect microphone or an external one).

Also, an additional activity was added in the Contemporary Music Composition game, to visualize the "augmented music score", as well as the errors with respect to the expert performance, both regarding to gestures as well as emotions. This tool is expected to facilitate the access to the knowledge of the expert, both gestural and emotional.

Furthermore, during this third period of the project, various improvements were made to the previously presented ICH games in D5.2. Besides, new features and new games were added to the platform. Furthermore, several improvements were introduced regarding the application-sensor communications.

All of the ICH games were designed with the help and close cooperation of ICH experts. For each game design, an original game scenario was prepared based on ICH experts' ideas and feedback. To avoid inconsistency between the games, all of the game implementations followed the same design pattern.

The eight ICH games are collected under a single application which also handles communication with different type of sensors. This single application not only provides a consistent interface and GUI units but also reduces the coding effort since many common functions are used by different games.

All games are based on a common structure, but there are differences with respect to technology, approach and the external sensors used in each game. Hence, the overall work exceeded the initial plan and caused a three months delay in the finalization of this deliverable. Moreover, many games require further testing and optimization, so final versions of the implementations will be delivered in Deliverable 5.7 – "Final Version of Integrated Platform", before they are ready for use in the demonstration and evaluation phase of the project.

2. Introduction

2.1 Background

"Intangible Cultural Heritage" (ICH) is defined as a part of the cultural heritage of societies, groups or sometimes individuals and it includes practices, presentations, expressions, knowledge, skills and related tools to all of these, such as equipment and cultural sites. ICH passes from generation to generation and gives people a sense of identity and continuity; it is the result of the continuous interaction of communities and groups with their nature and history and it promotes respect for cultural diversity and human creativity as discussed in detail, Deliverable D2.1 "First Report on User Requirements Identification and Analysis".

The main objective of i-Treasures project is to build a public and expandable platform to enable learning and transmission of rare know-how of intangible cultural heritage. The proposed platform combines lots of different technologies, like multisensory technology, singing voice synthesis and sensorimotor learning to leave the beaten path in education and ICH knowledge transmission.

In i-Treasures project, four main ICH use cases are selected:

i. rare traditional songs with the following sub use cases:
   a) Byzantine hymns,
   b) Corsican "cantu in paghjella",
   c) Sardinian "canto a tenore" and
   d) Human beat box;
ii. rare dance interactions with the following sub use cases:
   a) Calus dance,
   b) Tsamiko dance,
   c) Walloon dance and
   d) Contemporary dance;
iii. traditional craftsmanship focusing on the art of pottery making, and finally

iv. contemporary music composition, involving modern approaches of Beethoven's, Haydn's and Mozart's musical styles and interpretation.

In order to prevent these ICH expressions from extinction, we aim to provide a tool that will allow people to practice the ICH expressions. Towards this goal, game-like educational applications have been developed within Task 5.2. Multi-platform game-like applications for sensorimotor learning based on educational game scenarios that are provided in "D5.2: First Version of 3D Visualization for Sensorimotor Learning".

2.2 Aim of this Deliverable

This deliverable, named "D5.5: Final Version of 3D Visualization for Sensorimotor Learning", is the second major outcome of the Task "5.2: 3D Visualization Module for Sensorimotor Learning", which is part of "WP5: The Integrated Platform for Research and Education".

Within the context of this deliverable, we describe the final version of a 3D visualization for sensorimotor learning module, which introduces eight different game-like applications for the four ICH use cases that are listed above.

2.3 Report Structure

The structure of this document is the following:

- Section 1 is the executive summary.
- Section 2 is this introduction.
- Section 3 presents the general structure of the 3D visualization platform.
- Section 4 presents the updates of the game-like applications as well as new games introduced during the current (third) project period.
- Section 5 describes the generic framework for the creation of dance/body-motion-based game-like applications
- Section 6 describes the game analytics that are provided to the i-Treasures platform by the game-like applications
- Section 7 describes the web version of game-like applications, which can act as a demo for the provided applications, supporting the "Getting started" and "Observe" functionalities.
- Section 8 draws some conclusions and discusses future work
- Section 9 provides references and
- Section 10 provides user manuals for the game-like applications as an Appendix to this Deliverable

3. Structure of 3D Visualization platform

3.1 Design of the Platform

In this part we provide summary of design aspects. For detailed information please refer to Section 3 of D5.2 "First Version of 3D Visualization for Sensorimotor Learning". In this project, we decided to endow the sensorimotor learning module

with game-like applications aimed to support the learning and transmission of the considered ICH expressions. The game-like applications in this platform can be broadly considered as Serious Games for educational purposes. The main reasons for using such game-like applications for ICH safeguarding and transmission are:

- Games support constructive, experiential and situated learning,
- Games support procedural learning,
- Games are engaging and motivating,
- Games are able to promote self - regulated learning and
- Games support collaboration among the learners.

In designing the game-like applications we have followed design guidelines. As a first step we took pedagogical aspects into consideration such as educational objectives, content, play out of the situation and feedback. Similarly the design of the games' scenarios also followed a structured approach. Both the game environments and the player sessions are planned in a very detailed approach with the help of game designers, educational experts and domain experts. Similarly the evaluation and scoring approach, the data collection methods and the graphical user interfaceare all designed with the joint work of the same team.

3.2 System Architecture

Figure 1 shows the system architecture of the final platform. The initializer is responsible for activating sensor setups and starting the game-like application. At this step 3DVMSL communicates with WP3 modules. This action is followed by the login where the user credentials are taken from the Web Platform. Thereafter, the game menu is shown and the user can either see the game tutorial or select one of the activities offered, such as Activity 1, Activity 2,… or Final Challenge. In the Observe mode pre-captured data from expert performances are displayed to the user. The Practice mode allows the user to imitate the expert moves and evaluate his/her performance. Therefore, there is a bilateral communication between the 3DVMSL and the WP3 modules in order to transfer sensor data to the game.

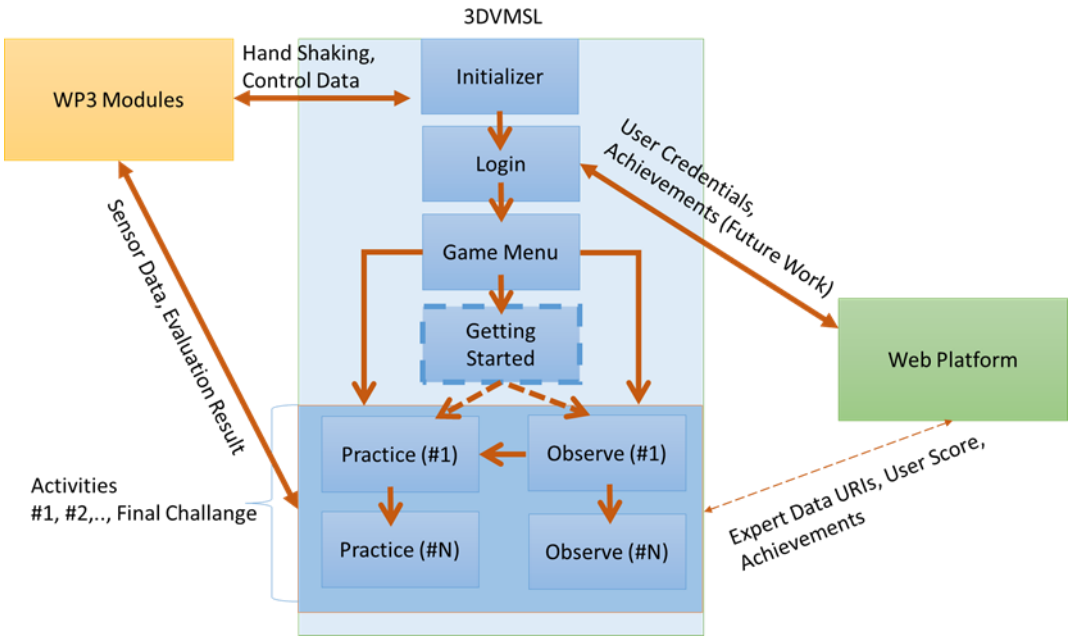


Figure 1. Detailed System Architecture of 3DVSML Module

Even though all the games use the same general architecture shown in Figure 1, each ICH use case has its own particularities regarding the sensors used, communication aspects, etc. The section below provides information regarding the differences in the architecture.

Dance applications communicate with Body and Gesture Data Capture and Analysis Modules (BGDCAM) developed in WP3. For every dance game this module shows differences considering the sensors included in the game. The details of the data capturing system can be found in section 3.1 of Deliverable D3.2 "*First Version of ICH Capture and Analysis Module*". Figure 2 shows the communication architecture of the dance games. As this figure shows both the Kinect v1 and Kinect v2 sensors (single or multiple) are used mainly. The BGDCAM controls multiple sensors and provides communication with 3DVMSL. All of the communication is done via TCP/IP messages. 3DVMSL also communicates with web platform via web services.

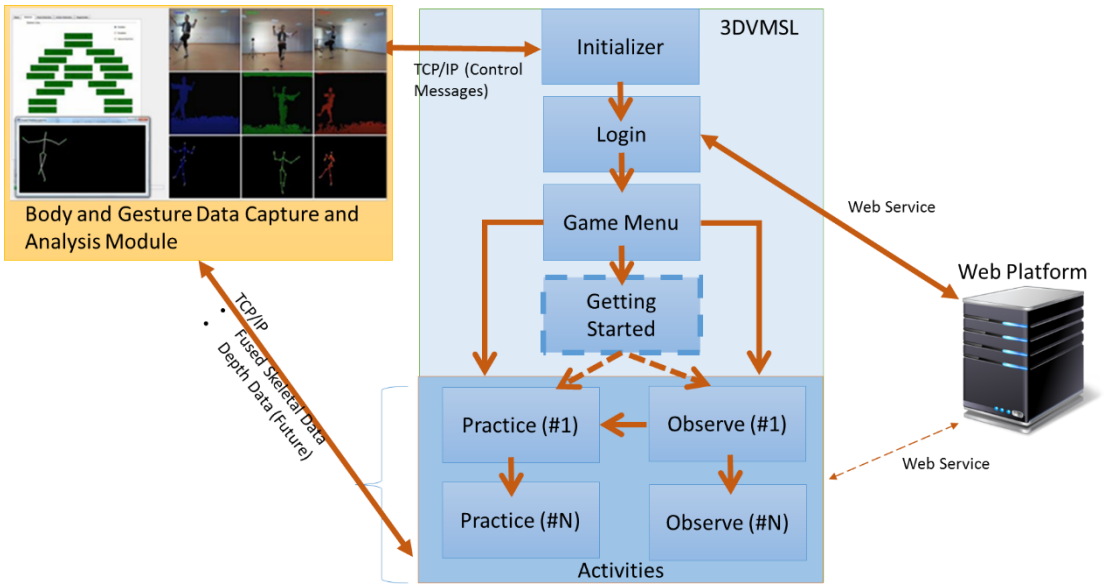


Figure 2. Communication Architecture of Rare Dancing Game-like Applications

Rare singing games (Human Beat Box and Byzantine Singing) aim to let the learner observe and practice the basics of HBB or Byzantine Singing by using various sensors and gaming interface. In these games 3DVMSL communicates with two modules of WP3 which are Vocal Tract Capture and Analysis Module (VTCAM) and Facial Expression Data Capture and Analysis Module (FEDCAM). The details for both modules can be found in presented in section 3.2 and section 3.4 of Deliverable D3.2 "First Version of ICH Capture and Analysis Modules". 3DVMSL also provides bilateral communication with web platform. Figure 3 shows the communication architecture.

The communication with VTCAM is done via RTMaps which is a software module for real time multisensory/multi-source applications, data acquisition, rapid prototyping and data fusion [1], 3DVMSL checks the availability of the sensors by sending and receiving simple control messages via TCP/IP protocol. In the Practice section, the 3DVMSL starts the system and gets multiple video streams for lips and tongue contour. It also gets vertices of the animated 3D tongue model. RTMaps also streams audio data of the learner to the external server via web service. This server provides the result of audio comparison to the 3DVMSL via another web service.

The communication with FEDCAM is realized with TCP/IP messaging. 3DVMSL gets facial data from the FEDCAM which uses a Kinect V1/v2 sensor.

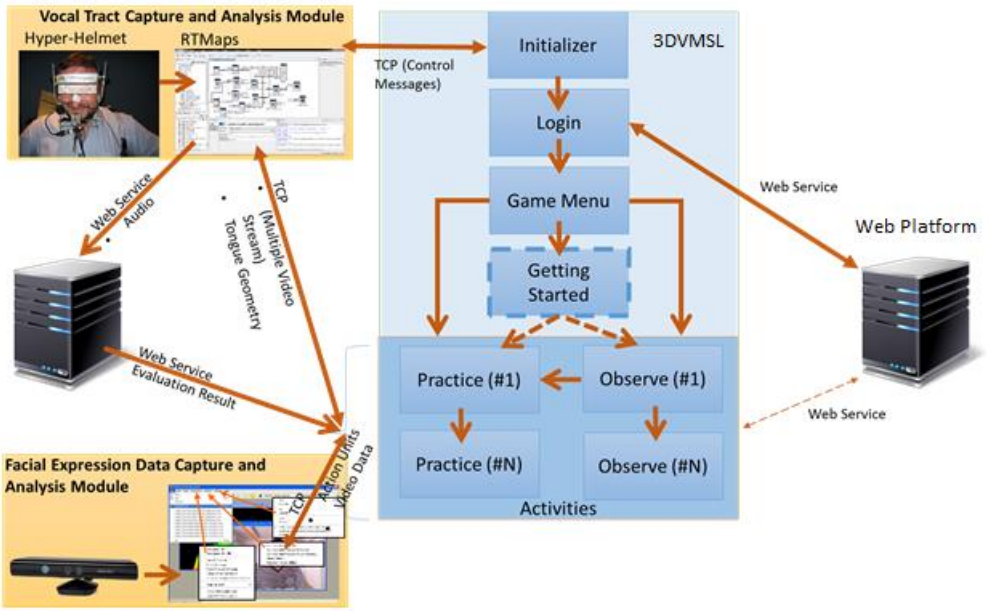


Figure 3. Communication Architecture of Rare Singing Game-like Applications

Rare traditional craftsmanship aims to let the learner observe and practice the basic moves of wheel pottery by using various sensors and gaming interface. Figure 4 illustrates the communication architecture. The initial definition of the data capturing system can be found in section 3.1 of Deliverable D3.2. BGDCAM performs a data fusion by using various sensors such as Leap Motion and Animazoo. The fused data are send via TCP/IP messaging. In the future version outline contour of the virtual pottery will also be transferred.

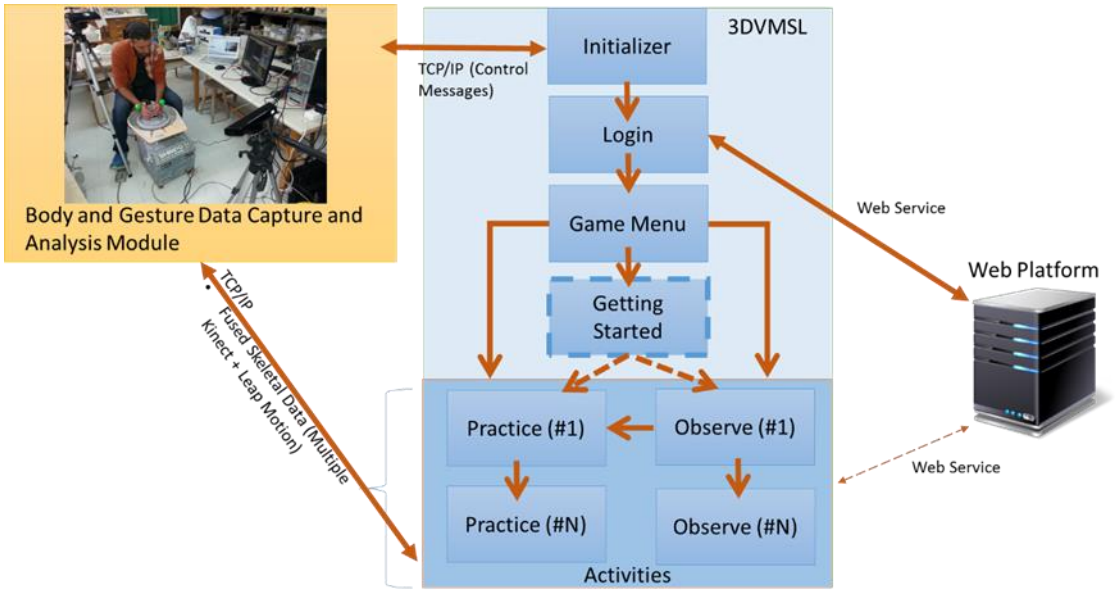


Figure 4. Communication Architecture of of Traditional Craftsmanship Game-like Application

The last game refers to the contemporary music composition use case of the i-Treasures project. In this game we started developing a novel intangible musical instrument, which is supposed to map natural gestures performed in a real-world environment to music/voice segments.

The Contemporary Music Composition game aims to let the learner observe and practice the basics of the new innovative intangible musical instrument by using

various sensors and gaming interface. In this game 3DVMSL is connected to the Max interface which is a visual programming tool for media. Therefore the application communicates with Max interface not directly with sensors. 3DVMSL checks the availability of the sensors by sending and receiving simple control messages via UDP messaging. Figure 5 illustrates the communication architecture. In this game, the learner uses Emotiv sensor, inertial sensors and Leap Motion sensor. In the practice section, 3DVMSL starts the system and gets skeletal data for animation. This is also done via UDP Protocol based bilateral messaging. Currently, Max plays the synthesized sound, which is driven by the moves of the learner. Similarly, the Max is used as an interface for Emotiv as well. Details on emotional activity capturing are discussed in section 3.3.3 of Deliverable 3.2 “First Version of ICH Capture and Analysis Modules”.

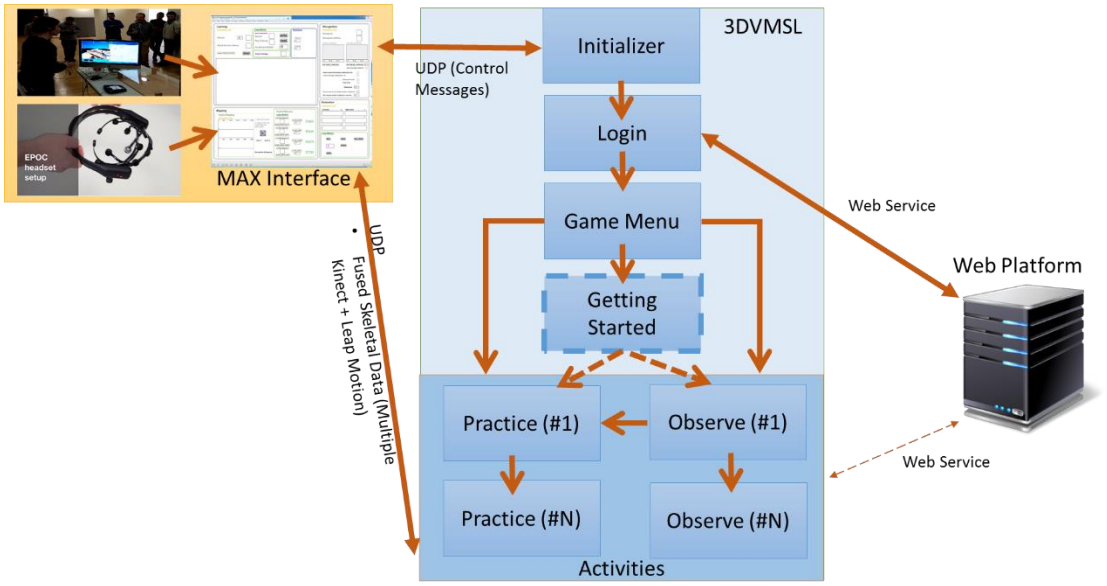


Figure 5. Communication Architecture of the Contemporary Music Composition Game-like Application

3.3 Use Case Diagram

Figure 6 shows the basic “use case scenario” defined for the main actor (the learner), which is applied for all the games. Although, each game has a different game scenario, they all follow this common structure. All games have one user: the learner who has the same accessibility to three basic functions (Login, Observe and Practice). The term “use case scenario” in this section refers to the list of user actions/interactions with the system. The learner starts with “Login” and either selects an ICH use case or changes some game setting(s). He/she can then select an activity or examine the “getting started” section. Afterwards, the learner plays the available activities/sub-activities or tries the “final challenge” of the game.

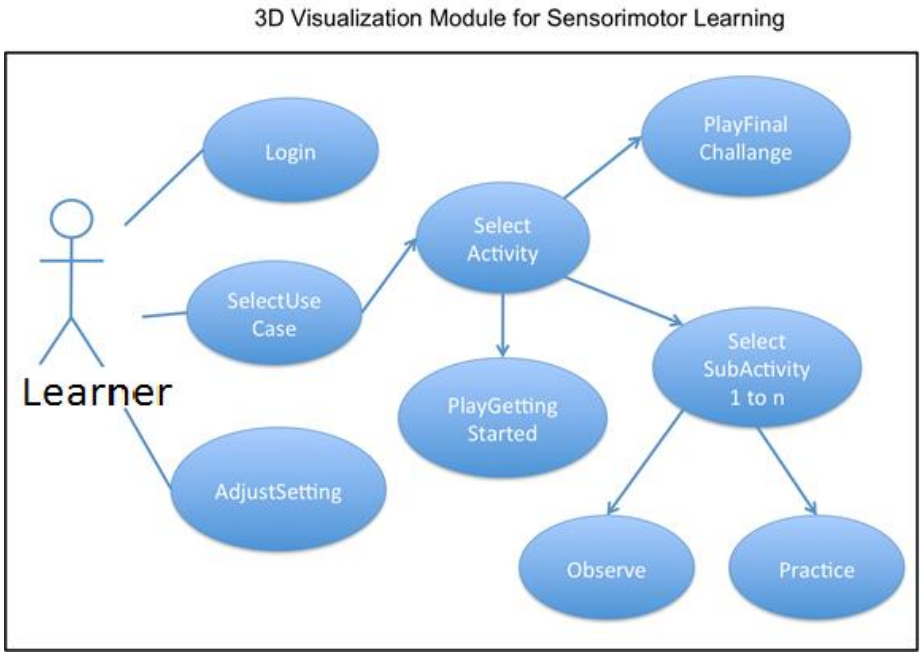


Figure 6. Use Case Diagram for actor Learner

3.4 Game Interface

As already mentioned, all of the games are combined in a single framework. Therefore the user is expected to select a game by using this interface. The interface stars with a splash screen that shows the project logo of i-Treasures. Following the loading, the interface prompts a login screen with multiple language options. In this screen, the user can log in as a registered user or a guest. Registered users are checked by using the web service provided by web platform. After a successful login, game selection screen is presented to the learner, and the learner is allowed to select any of the four use cases of I-Treasures project, and this operation is followed by a similar screen where the learner picks any sub use-case (such as Tsamiko, Calus or Walloon in rare-dancing use case). As a next step, the learner gets access to the main screen of the game, presenting the options shown in in Figure 7. At the beginning, the user is allowed to choose the “Getting Started” or the Activity 1. If the learner completes successfully the Activity 1, she unlocks the following activities and is now free to select the unlocked activities, both the “Observe” and the “Practice” option.

Figure 7. Game Menu Screen

The functionalities and detailed features of the game menus and the other screens are provided in Section 3 of D5.2.

4. Game-like applications

4.1 Rare dancing

4.1.1 Tsamiko

4.1.1.1 Introduction

Tsamiko game is for learners who want to learn Tsamiko dance by observing experts' movements in the Observe mode and by reproducing these dance steps recorded by a Kinect v1 or Kinect v2 camera in the Practice mode. The game software guides the user through activities and teaches him/her how to dance Tsamiko properly. The game-like application and its assets were also presented in detail in Deliverable D5.2 "First Version of 3D Visualization for Sensorimotor Learning".

4.1.1.2 Assets

In order to develop the Tsamiko dance game, we prepared a virtual tutor (2D Character), 3D avatars of the expert and the learner, 3D environments for the game.

The role of virtual tutor is to navigate and help the student by giving oral and written feedback during the game to improve the learner's performance. Based on the performance of the learner, the mood posture (happy, explanatory, unsatisfied, neutral and satisfied) of the avatar is automatically selected by the system.

We have prepared 3D animatable virtual characters to represent both the expert and learner avatars. We tried to use traditional clothes for the expert and casual clothes for the learner in order to differentiate the characters in the game. To animate the 3D

avatars, we combined both the inverse and forward kinematics approaches to have a better and more natural motion.

For the initial two activities of Tsamiko game, we prepared a modern dance studio with parquet floor and a big mirror at the back wall. For the final challenge activity, a 3D model of the famous “Odeon of Herodes Atticus” in Athens is designed, which is provided as a motivation factor and can be unlocked only when the learner completes the previous activities.

4.1.1.3 User performance Evaluation

Tsamiko game uses Kinect input (v1 or v2) to capture learner data and compares the inputs with expert data in order to generate an evaluation and to provide a feedback to the learner. The data is captured and processed by the Kinect Acquisition Tool explained in D3.2 “First Version of ICH Capture and Analysis Module”. Therefore the input data is transferred to the game PC by using a TCP/IP based network communication tool after bone mapping and filtering is applied.

In order to provide feedback, moves of the learner avatar and expert avatar are compared using a Dynamic Time Warping (DTW) combined with a Fuzzy Inference based algorithm. DTW is a well-known technique for measuring similarity between two temporal sequences that may vary in time or speed. We used various feature sets which are used as input to DTW separately in order to obtain distinct distance measures. Taking into account that in Tsamiko dance the leg movements constitute the key elements of the choreography, knee and ankle joint positions were used. For each feature set, DTW provides a distance measure between the time-series of the user and the expert. These distances, one per feature set, are subsequently fed to a Fuzzy Inference System (FIS), which computes the final evaluation score. The evaluation function produces a normalized scalar value between 0 and 100, which can also be translated into the appropriate text to be displayed by the virtual tutor.

In addition to the final score, displayed to the user after the end of each exercise, the above algorithm is also used to report the instant score, which is the correctness of the motion during a specific time segment. This is performed by implementing a time window over the recording of the expert and comparing it to the performance of the user keeping only several last (100) frames of his motion. This is reported periodically to the user, and helps him get a feedback about his current performance.

With this feedback he/she can understand that if his/her moves are correct or if they require any alteration. In addition, a colour coded scale is provided (**Figure 8**).

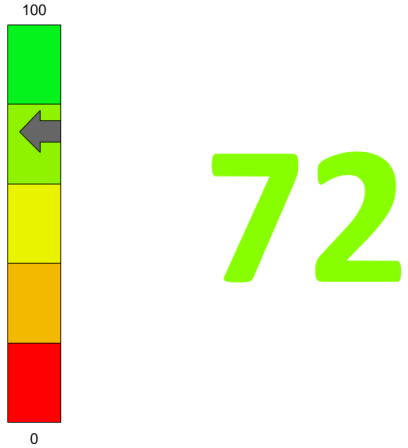


Figure 8. Colour coded scale and numerical score value

4.1.2 Calus

4.1.2.1 Introduction

Romanian Căluş originated as a healing and fertility ritual performed by groups of an odd number of men, bound together by an oath. By the beginning of the 20th century its ritual form survived mainly in southern Romania and among Romanian minorities in northern Bulgaria [5], although remnants of this custom could be found in much of the rest of Romania, and throughout the Balkans.

Similarly to the Tsamiko game, the Calus game allows users to learn Calus dance by observing expert's' movements in the Observe mode and by reproducing the dance steps recorded by a Kinect v1 or Kinect v2 camera in the Practice mode

4.1.2.2 Assets

A 3D avatar of the Calus dancer with the traditional clothing is prepared and added to the game to give to the learner the impression of a real Calus dancer (Figure 9). Also, a 2D virtual tutor was created to guide the learner. This tutor wears traditional clothes. As in the Tsamiko case, it has different mood postures (happy, explanatory, unsatisfied, confused etc.), so as to provide the learner with the proper feedback. The same dance studio used for the Tsamiko game is used as environment background. A video of an expert performing the target move is shown on the left of the game screen to show the learner a real-life Calus dancer.

Figure 9. Calus dancer avatar

4.1.2.3 Visualization

To visualize the expert's and learner's 3D avatars, inverse and forward kinematics are used. Specific joints such as wrists, elbows and knees are selected beforehand and are added to a kinematic chain as starter joints. When the Capture (WP3) module sends data to the game, values are set to starter joints. Other joints' positions

and orientations are then calculated via the kinematic system, to get the desired pose of the dancer, whilst respecting anthropomorphic limits, thus avoiding the creation of non-realistic body postures.

4.1.2.4 User Performance Evaluation

To evaluate the Calus dance, the Tsamiko evaluation system is used. The Game module and Capture (WP3) module are connected via TCP/IP and all input data received from one or multiple Kinect v1 or v2 sensors re-streamed to the game.
While streaming these data, the capture (WP3) module also stores them, compares them with the experts' moves, to evaluate the learners' movements and provide scores (0-100) to the game module, as an instant feedback to the learner. As in the Tsamiko game, the result is presented to the user both either using a colour coded scale and a numerical score value.

4.1.3 Walloon

4.1.3.1 Introduction

Walloon traditional dances are essentially peasant dances originated from the 18th, 19th and early 20th centuries and practiced in the Walloon region of Belgium. They were originally mostly danced in popular balls in the villages, but almost disappeared at the end of the 19th century and the beginning of 20th century. A few people and groups interested in preserving and perpetuating this intangible heritage conducted dance collections at their own initiative, by interviewing older people who used to perform the dance and, hence, were living representatives of this heritage, or found information in notebooks from ménétriers (dance leaders), who used to go to local events (weddings, etc.), and in villages to play music and animate the traditional balls.

In the Walloon dance game, the expert's movements are captured by a high precision motion capture system (Qualisys), but as this system is very expensive, markerless sensors such as Kinect v2 are used for playing the game. For evaluation, the change of data quality has been taken into account by creating a sensor-dependent gesture recognition and evaluation system. The algorithm used for recognition and adaptation is described in D3.3 "Final Report on ICH Capture and Analysis".

4.1.3.2 Assets

Differing from other dance games, Walloon game is played in a rural environment. A 3D environment is modelled after a Walloon region of Belgium. In the environment there are traditional stone houses and a church inside a green field with trees. Furthermore, the 3D avatar is modelled after a traditional Walloon dancer. As in the other games, a virtual tutor is developed to give feedback to learner.

Figure 10. Walloon game environment

4.1.3.3 User Performance Evaluation

For the Walloon dance, stylistic analysis and comparison is used for the evaluation of this dance. This game uses Kinect v2 to capture the performer's gestures and the evaluation is made using Hidden Markov Models as described in D4.5 *"Report on ICH Indexing by Stylistic Factors and Locality Variations"*. A dynamic programming algorithm called Viterbi is used to decode dancers' performance. Time normalization of log-likelihood by dividing the length of the sequence is executed. A percentage score to the user is given. This can be interpreted by the user. A good performance is higher than 75%, a medium performance is between 75% and 50% and a bad performance is below 50%. In the Walloon dance, errors usually occur because learners don't cross their feet and they don't bend their knees enough according to experts. Hidden Markov Models used to evaluate the dance moves estimate a global score of a performance relative to evaluation models. Without extra detail, the given score represents an overall evaluation of the students' performance. To have more detail, of the students' performance of the dance, such as whether the student is doing the moves slower or faster than the expert or if s/he is not bending his/her knees enough, statistical factors, as well as the feature exploration module, also described in the deliverable D4.5, can be used.

4.2 Rare singing

4.2.1 HBB

4.2.1.1 Introduction (TT)

The Human Beatbox (HBB) Game was exhaustively described in D. 5.2. The game involves producing rhythm and drum beats using the vocal tract. As facial and oral-movements (like tongue, lips and jaw) are involved in HBB, the expert's data are shown accordingly. In this game, the learner is first guided through basic descriptions of the sound types to be produced. As seen in Figure 11, the ultrasound video of

mouth and tongue of the expert is displayed on the right. It is accompanied by another video, which shows camera images of the expert's mouth and a 3D model that shows the reconstruction of the tongue of the expert. These videos and models can be switched off by clicking on them. The user is also able to modify the size and position of these windows.

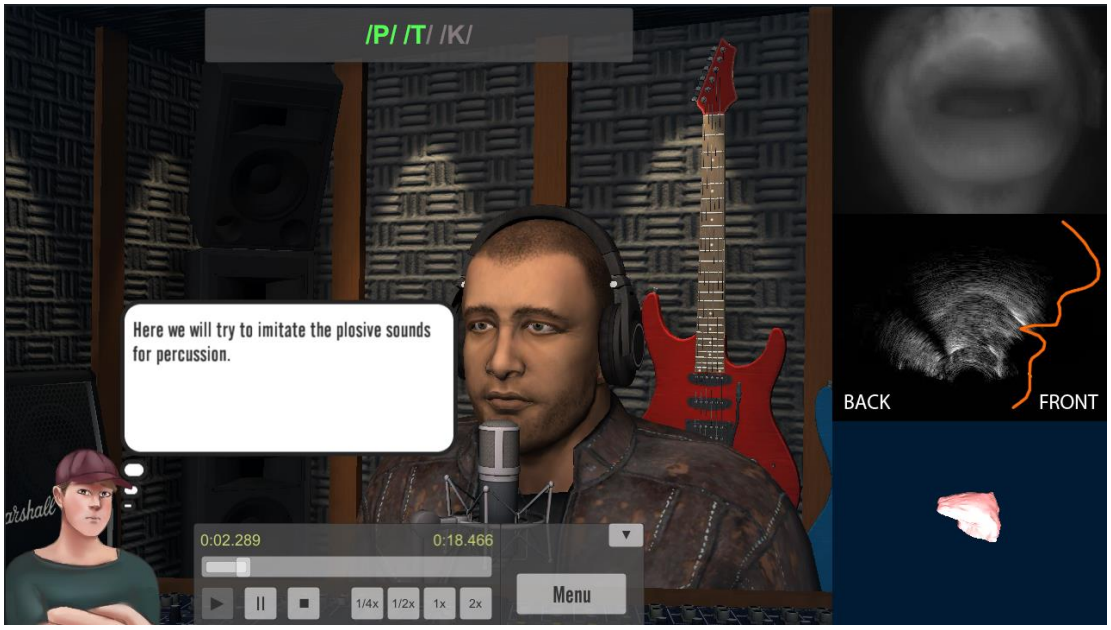


Figure 11: Screenshot from HBB Game

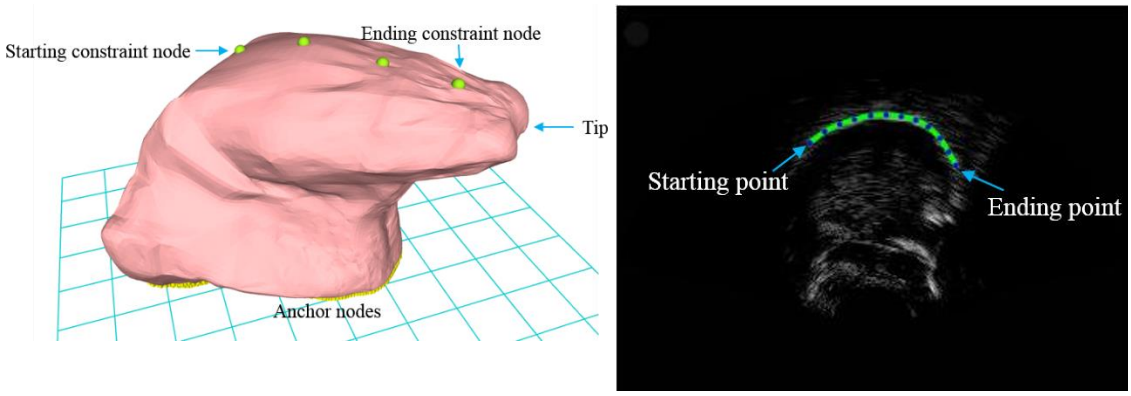
The visualization module must provide relevant information on the production of the sound to the user. One important part of this sound production is the configuration of the vocal tract. In i-Treasures platform, an ultrasound probe is used to gain information on the mouth cavity. The main salient feature of these images is the upper contour of the tongue. This can be used as it is since the user should be able to extract this contour and compare it to a reference image provided by the expert, but a 3D model of the tongue could help the user to move his tongue in the correct position. It is thus needed to establish a correspondence between the contour of the tongue extracted from the US images and the nodes of the 3D model used to move it. One of the challenging points of this task lies in the fact that US images cannot provide information on fixed points (named "tissue points") on the surface of the tongue since the tongue is an elastic material. Thus it is not possible to register tissue points of the tongue to fixed nodes of the 3D model which are used to move the model. This is a major problem because the 3D model is highly sensitive to the displacements of these nodes and incorrect displacements can lead the 3D model to behave in an unrealistic way. To address this issue, we propose to build a database of realistic 3D model configurations, to extract the mid-sagittal contour of these configurations and then to compare the current contour extracted from the US image to all the contours in the database. The positions of the nodes used to move the 3D model will then be the ones of the closest contour in the database. This ensures that the configuration of the 3D model is realistic.

More details can be found below.

4.2.1.2 3D Tongue Visualization (UPMC)

The method we used for improved 3D Tongue Visualization can be divided into 4 steps :

Step 1: Initialization. Four constraint nodes are selected manually on the 3D model (Figure 12a). The first and last nodes are associated to the starting and ending points of the contour extracted from the 2D US image (Figure 12b).



(a) (b)

Figure 12. Elements used for the 3D visualization. (a) The 3D model used in our framework, the green circles denote the constraint nodes, whose displacements are associated with the modal displacement. The yellow nodes are anchor nodes whose displacements are zero during the deformation of the tongue model. (b) Target curve extracted from the image, the green lines are the surface of the tongue.

Step 2: Database construction. A database of 3D tongue shapes is built by assigning random, limited displacements in the mid-sagittal plane to the four constraint nodes. In our experiments, 1000 shapes were used. For every 3D tongue shape, a contour is extracted by using the nodes lying on the surface of the tongue between the starting and ending constraint nodes in the mid-sagittal plane. This database can be constructed in a more sophisticated way by creating shapes sampling the real shapes of the tongue during song production. Different databases could be built for different types of songs, since, for example, Human-Beat Box uses very unusual tongue positions.

Step 3: Contour extraction. An improved active contour based method is used to extract the contour from the US image (Figure 12b).

Step 4: Similarity measurement. The similarities between the US contour and the 3D contours of the database are evaluated using the Mean Sum of Distances (MSD), based on the distances between the points of each contour. A penalty term is added to filter out unphysical local deformations. The most similar 3D tongue shape is selected to represent the target curve shape associated with the ultrasound image.

The advantages of this method are multifold; first, the 3D shape associated with the US image will be realistic, secondly, measuring the similarity between 2D curves is more efficient than comparing US images. Finally, although tongue contour extraction has still issues, it is much more robust on US images than tissue points tracking methods.

Results

This method has been tested on some vocalizations and has shown promising results. An example is given in Figure 13.

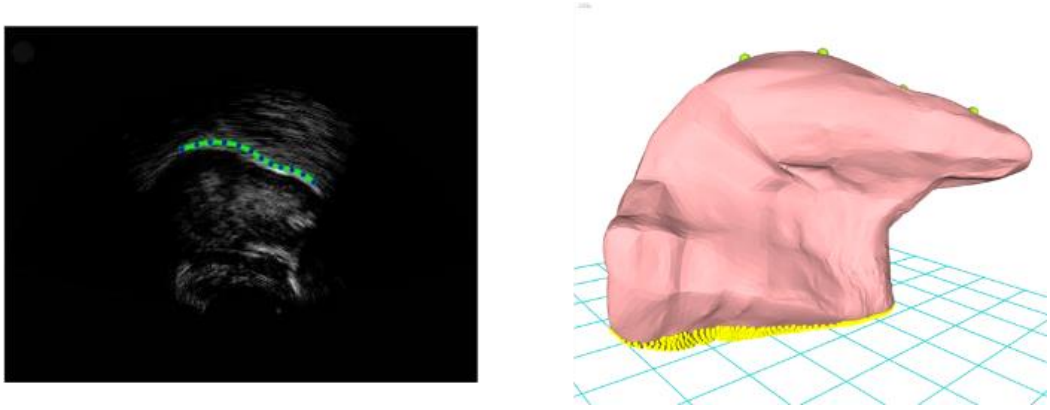


Figure 13. Sample frames of 3D tongue modeling. The meaning of the color line and points is the same as Figure 12. The 3D tongue shapes are selected from the 3D tongue database.

Perspectives

Several directions can be followed in order to improve our method. First as said above, the construction of the database can be refined. Secondly MSD is a rather crude similarity measure and more specific measurement may need to be developed. Furthermore, this method can only take into account mid-sagittal motions and additional information should be extracted from the US image in order to infer out-plane motions. Lastly the use of other imaging modalities such as EMA that gives the displacement of tissue points would be of great interest to evaluate the performance of our method.

This work will be presented the 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016) to be held during 20-25 March 2016. Please refer to paper #3055 for more information.

4.2.1.3 User performance evaluation (Real time evaluation) (UMONS)

Learner performance evaluation for HBB has been described in deliverable D5.2 (at the end of Year 2), and more recently updated versions in deliverables D3.3/D3.4, at the end of Year 3. HBB practice is heavily based on the production of vocal imitations of real instruments, so the evaluation method is designed for measuring the distance in timbre between the student and expert realizations of these sounds. For this, we first used HMM recognition. Although HMM-based HBB recognition has been significantly improved in Year 3 (see D3.3), the sensitivity of the approach still makes it unsuitable for scoring. One reason is that the statistical model over-fits to expert sounds because currently, although unique, the database for training contains the voices of two experts only. Hence, we then rather used DTW alignment, with a distance metric between acoustic features tuned to provide relevant local “scorings”.

The game enables the student to practice a cappella, as the warping/alignment procedure through DTW makes it possible to match student and expert sounds for computing proper local and vocal event scorings. However, in that case, scoring results are only provided after the student performance is finished. We are hence currently working on a real-time approach in order to improve on this. For enabling real-time feedback, the approach is to enable the student practice in a sing-along mode, where he sings (preferably using headphones) together with a metronome and expert singing in background. In that case, warping/alignment has to be performed

only very locally (to tolerate for small timing deviations of the student), and evaluation can be provided almost instantly for each sung sound.

All the analysis and scoring system has been implemented as a standalone server to which the game sends http requests, and which returns scorings for each vocal event as a response. The game embedding this functionality has been demonstrated.

4.2.2 Byzantine

4.2.2.1 Introduction

The Byzantine Music game is similar to HBB game, in that the user learns to sing hymns instead of beatbox patterns. However, the HBB game makes use of the expensive ultrasound device, a fact that severely limits the possibility to market this application for wide use. Thus, in the Byzantine music game, the idea is to use only very low-cost sensors, such as a Kinect device and a microphone (either the Kinect microphone or an external one).

Learner performance evaluation is provided by a performance evaluation server developed by UMONS that receives audio data from the learner performance from the game application. Regarding user performance evaluation two modes are defined: either during (“real-time”) or at the end of the performance (“off-line”). In real-time mode, the user can see the waveform of his/her performance and compare it to the corresponding waveform of the expert.

The game screen also provides the expert avatar, whose face tracking data is demonstrated. Therefore, the learner can imitate the mouth of the expert. Furthermore, the video (and audio) of the expert performance is provided, while the lyrics and music score of the hymn are also provided in Karaoke-style videos, to better guide the user.

4.2.2.2 Assets

3D environment

The 3D Environment asset is the 3D model of the sample church pictures sent by the partners. Three different views generated from the 3D model of the church are provided in Figure 14, Figure 15 and Figure 16. The church model is also designed to reflect the game culture.

Figure 14: 3D Environment for the Byzantine Game – Scene-1

Figure 15: 3D Environment for the Byzantine Game – Scene-2

Figure 16: 3D Environment for the Byzantine Game – Scene-3

3D Learner Avatar

The 3D Learner Avatar, given in Figure 17, represents the person playing the game. During the game, the avatar will stand between the microphone and the church 3D Environment, and move his/her lips based on the input data.

Figure 17: 3D Learner Avatar for Byzantine Game

Expert recording, lyrics and music score

Video recording of the expert took place in a sound recording studio on UOM's premises, organized by UOM in cooperation with CERTH, on February 2016. The video (Figure 18) helps the student to imitate the experts' movement (tempo), expressions and passion and was captured in HD definition with a Kinect v2 sensor. Facial data were also collected using a Kinect sensor with the appropriate software.

Figure 18 Expert video recordings

Furthermore, a set of videos presenting the byzantine music scores (partitura), i.e. the sequence of "Neumes" (Interval Signs), were created using specialized software. Screenshots from these videos are presented in Figure 19. The current Neume that the expert sings at each time instant is highlighted, while the audio from the expert is also included. The score that is presented to the learner starts with the first notes (tonality), followed by some rests with metronome that show the tempo and then the Chant begins.

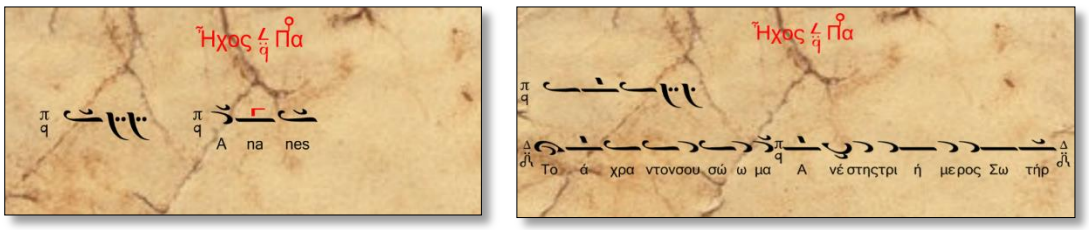


Figure 19: Screenshots from videos of Byzantine music score

The Byzantine music scores that we selected for the Byzantine game are:

1. First mode's Apechema (3 tones) - tempo 44 bpm

2. Scale of First mode (3 tones) - tempo 50 bpm
3. Parallage: a) Rhythmic Byzantine notation – tempo 52 bpm b) Melodic Byzantine notation (3 tones) – tempo 70 bpm
4. a) Rhythmic Lyrics tempo 61 bpm b) Melos (in Greek “Μέλος” - Melody with lyrics/Chant) (3 tones) tempo84 bpm

The final data that were provided for the development of the game are:

1. The recorded audio of the expert (.wav files)
2. The Byzantine music score videos (including recorded audio and metronome sound)
3. The recorded videos of the expert (.avi files)
4. The sound of the metronome (.mp3 files)

Byzantine Game User Interface

A screenshot from the Byzantine Game User Interface is illustrated in Figure 20. Although the metronome sound currently exists in the music score videos, a video metronome will also be added to indicate the beat of the music. The metronome can be configured or turned on/off during the game. Also, a spectrogram will be used to display the pitch values of the learner and expert performances as two different-coloured waveforms. In off-line user evaluation mode, only the expert’s voice waveform is displayed, since the recorded voice of the player is sent and evaluated only at the end of the performance. In real-time user evaluation mode, both expert and learner’s waveforms will be displayed.

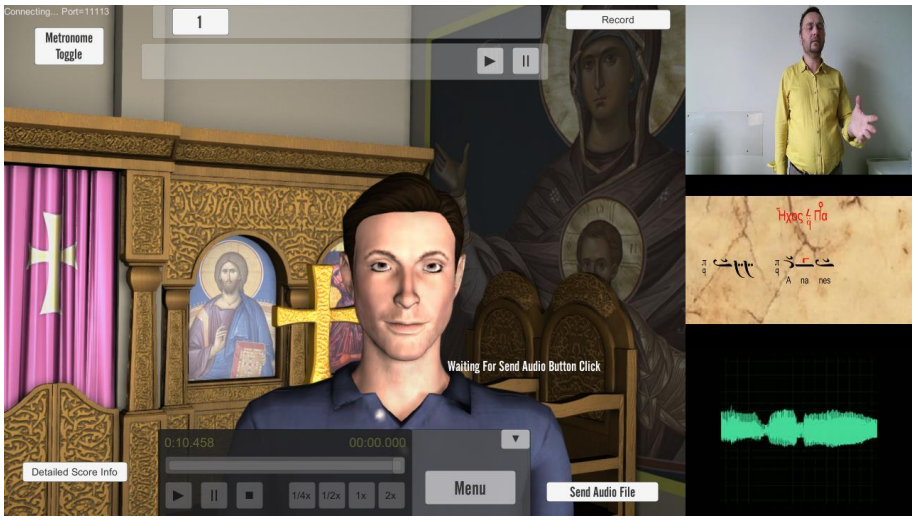


Figure 20: Byzantine Game User Interface

4.2.2.3 Performance evaluation

Byzantine Music game uses Kinect sensor to track learner’s face. A 3D avatar with face blend shapes is updated accordingly. Therefore, the user can track whether his/her mouth and face movements are similar to those of the expert. Learner performance evaluation is provided by an performance evaluation server developed

by UMONS that receives audio data from the learner performance from the game application. There, the pitch of the user's performance and the expert performance is compared and the learner is notified accordingly whether he/she sings the right note or not. Audio is recorded by a microphone, which is bundled by the Kinect sensor.

Learner performance evaluation for Byzantine music is relying on pitch analysis (see. deliverables D3.1/D3.2 at the end of Year 1 and D3.3/D3.4 at the end of Year 2). The evaluation method hence needs to be based foremost on measuring the distance in pitch between the student and expert realizations (or musical score to be sung). However, one has to consider specific aspects of byzantine music. First, byzantine music does not rely on an equal-tempered diatonic scale. Second, the singing style makes heavy use of ornaments and appoggiaturas. In a learning schedule, one may want first to evaluate a student based on the correctness of note pitches (and durations) disregarding the fact that he is not producing appoggiaturas at the exact time instants than the expert (or even not producing appoggiaturas at all). In a mode where the scoring is done in comparison to an expert performance, it hence become necessary to filter out the impact of appoggiaturas, which if not would have a very large impact (as appoggiaturas can extend to 3 half-steps up, and have a significant duration compared to the total note duration). The scoring system hence embeds a pitch post-processing step that filters out the appoggiaturas (both from the student and expert) as well as the vibrato. Then, it also quantizes the pitches to a byzantine scale (for the expert, enabling the system to output a musical score from the expert performance). The post-filter relies on an adaptive morphological filtering, as simple media filtering did not work well. A side benefit from the approach is that is also filters out spurious pitch estimation errors. The method has not been published yet.

Just as for HBB, two modes should be considered. On one side, we have a cappella singing, which requires alignment of student and expert, implemented in the system. We are also currently working on real-time. The pitch post-filtering will entail some latency however.

For the technical point of view, the integration within the game relies on the same approach than HBB, with a server providing the analysis and scoring functionality through http requests/responses.

The rest of this section shows an illustration of running the algorithm on expert and student performances. The student performance is analysed using acoustic feature extraction (in Figure 21 only the spectrogram is shown).

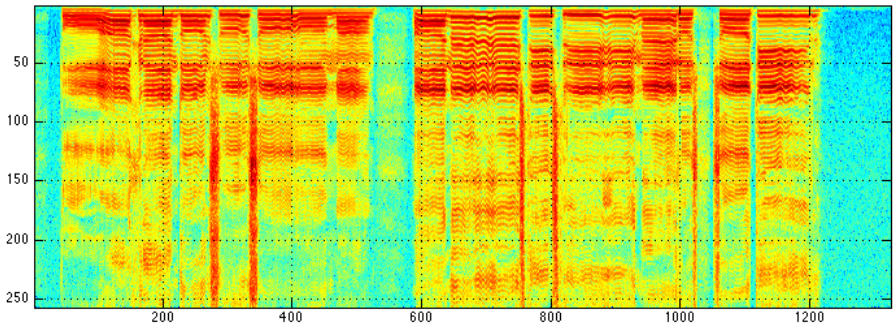


Figure 21: Spectrogram from the student performance

The expert performance is analysed similarly (Figure 22).

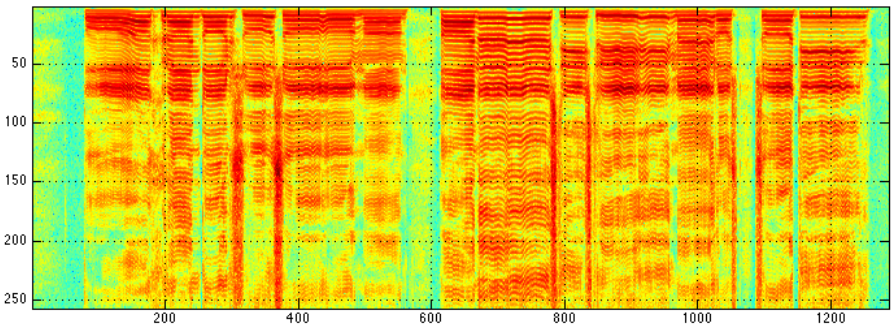


Figure 22: Spectrogram from the expert performance

Both performances are aligned, taking into account both cepstral acoustic features, and pitch (Figure 23).

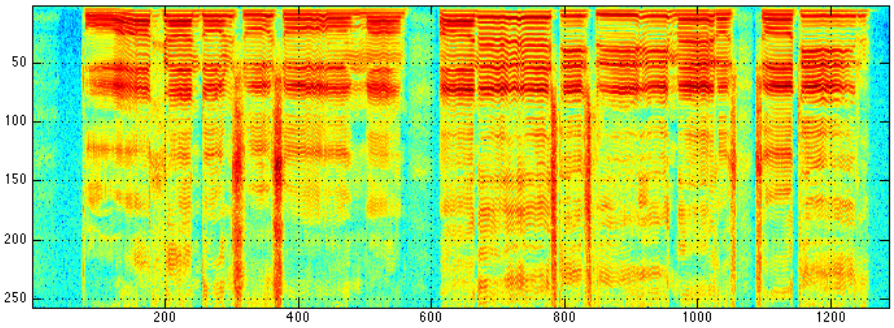
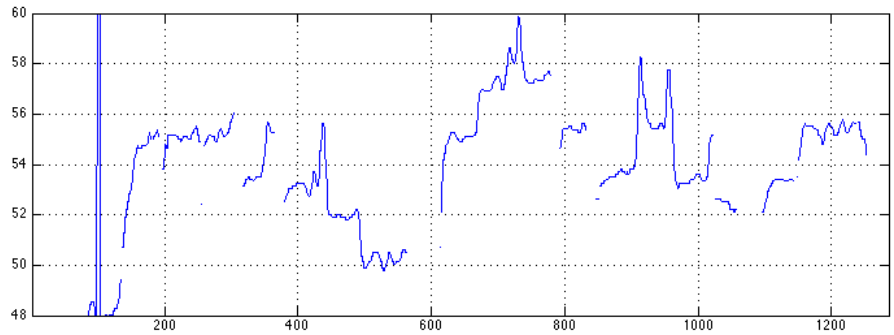
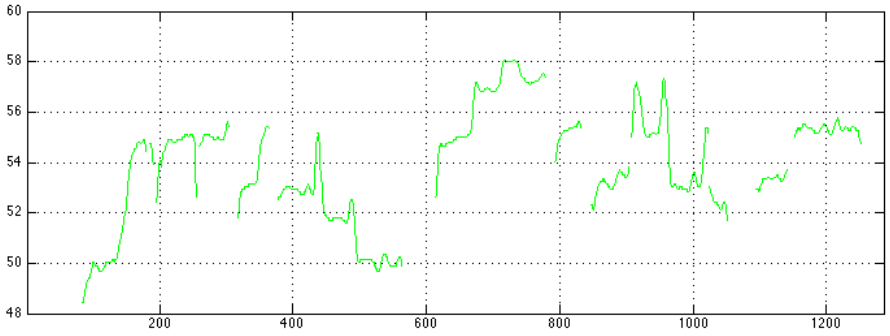


Figure 23 Aligned student performance

The pitch curves of the student and of the expert are computed (Figure 24). The scale of the y-axis of the shown figures hereunder is displayed in MIDI note numbers. Here we see notes from 48 to 60, which means from note C4 to C5. (Note there is a small pitch estimation error in the first note of the student performance, this does not matter a lot as it is going to be filtered out later and won't affect the scoring). The figures also show several appoggiaturas, as short and high pitch "overshoots" in between notes. Several notes also exhibit a strong vibrato (the last note in particular).



(a)



(b)

Figure 24: Pitch curves of (a) student and (b) expert

In the Figure 25, the two pitch curves have been superimposed. Also, the graph shows in tick lines the resulting pitch curves after post-filtering, again for both the student (in blue) and expert (in green). From this, notes can be extracted and each note of the student performance can be given a scoring. In this illustration, we can clearly see that the first note is not correct as it is sung 2 half-steps below the expert note. The error is here shown highlighted in red.

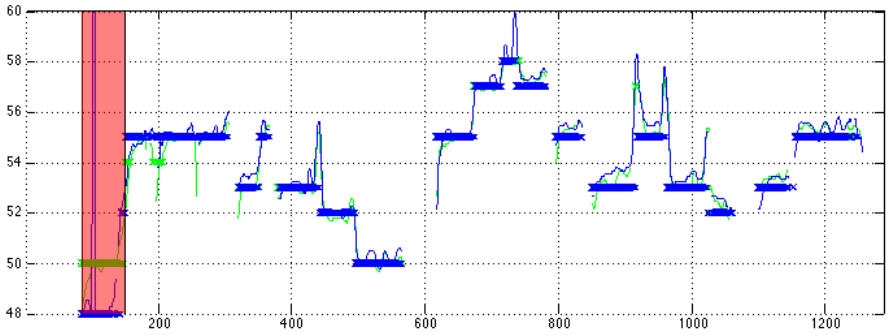


Figure 25: Superimposed pitch curves and learner scoring

4.3 Pottery

4.3.1 Introduction

Pottery game-like application focuses on the traditional craftsmanship use case of i-Treasures project. Wheel-throwing earthenware pottery was taken into consideration while designing and developing the traditional craftsmanship game-like application. There are four activities, which are different from each other and have different levels of complexity. The first activity focuses on throwing and centering the clay on the wheel, the second on how to make the bottom of the object, the third activity shows how to shape formation with a tool and the forth how to cut and remove the final object from the wheel.

A “virtual pottery” game-like application, based on hand and finger motion data captured by Kinect and Leap Motion sensors, has been developed in the previous period and described in Deliverable D5.2. When playing this game, the learner makes virtual gestures to create a virtual object without using clay (the gestures of the expert are also virtual). During this period, a new version of the pottery game has been developed, which allows the user to form real pottery objects using an actual potter’s wheel and clay. A PMD camera, Animazoo inertial sensors and Kinect cameras have been used to capture both the object and the potter’s movements.

This scenario presented significant challenges for hand/finger and clay visualisation caused by occlusions, difficulty in clay/hand segmentation, etc.). After extensive testing of different sensor technologies and setups, Animazoo technology turned out to be the most reliable and stable solution for capturing and visualizing the potter's hand movements. Regarding the visualization of the object, a tool to synthesize 3-D object shapes using parametric curves that are defined by control points has been developed and is presented below. However, more effort towards this direction is required to accurately visualize the clay.

4.3.2 Avatar visualization

The user data for the visualisation of the upper part of expert's body has been captured with Animazoo IGS 180 Synertial suit covering the upper body (wrists and head included) and containing 11 sensors (accelerometers, gyroscopes, magnetometers). The data thus includes information about the rotations in 3 axes (XYZ) of 11 body segments. An important advantage of this technology and consequently of this data set is that it is occlusion independent. Capturing the upper part of expert's with computer vision would be extremely difficult because of the possible occlusions of his hands, while working with the clay. The data is recorded in BVH files (Bounding Volume Hierarchy) following a very precise hierarchical structure.

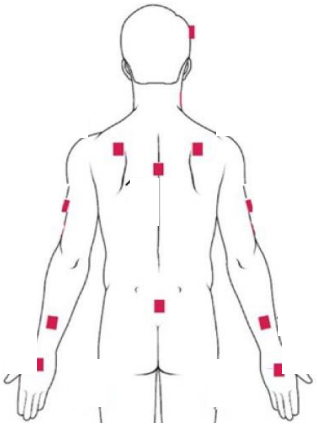


Figure 26. The dataset contains information on the rotations of 11 body segments

4.3.3 Pottery object visualization

In this section, the efforts towards modeling and visualization of the clay object are described. A software for manual modeling of a symmetrical 3-D pot object using only four 2-D control points and the efforts towards real-time object visualization using data from a Kinect sensor are also presented.

4.3.3.1 Pottery object modelling and offline 3-D pottery animation

A 2D cubic polynomial is rotated 360° to form the 3D pottery object. Cubic polynomial can be represented with 4 term coefficients (T_1, T_2, T_3, T_4) such as below equation;

$$P(x) = T_1x^3 + T_2x^2 + T_3x^1 + T_4x^0$$

To achieve manual pot object visualization, Bezier curves are used. A cubic polynomial can be generated with using a 4 2D control point Bezier curve. So, with using only 4 2D control points, a pot object visualization is achieved. Let us name

these control points P_1, P_2, P_3, P_4 where $P = (x, y)$. Using those 4 control points, any 2D point position ($B(t) = (x_t, y_t)$) at normalized value t on this cubic polynomial curve can be calculated as stated below;

$$B(t) = (1-t)^3 P_1 + 3(1-t)^2 t P_2 + 3(1-t) t^2 P_3 + t^3 P_4 , \qquad 0 < t < 1$$

While translating the control points vertically only, it is possible to prevent the horizontal movement of each control point and fixing them at the uniform x coordinate intervals as in Figure 27 below.

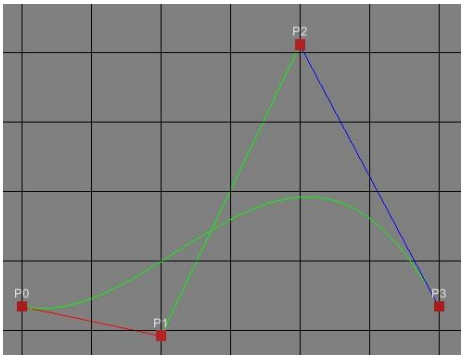
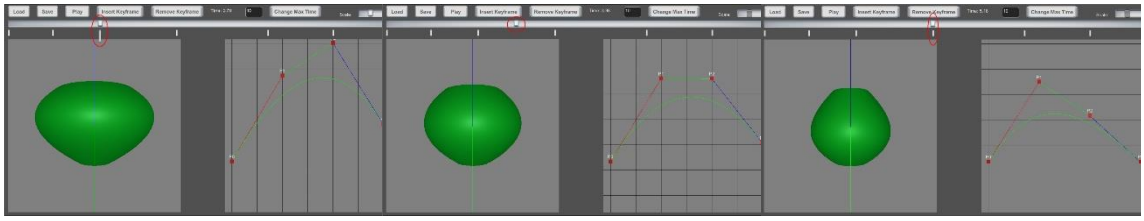


Figure 27: Bezier curve with uniform x coordinate intervals

By doing this simplification in above figure, x_t value between control points can be linearly calculated. At the same time, stating the formula of y_t value is enough to calculate any $B(t)$ point. Formula of y_t is stated below where y_1, y_2, y_3, y_4 are the y components of the control points P_1, P_2, P_3, P_4 consecutively;

$$y_t = (-y_1 + 3(y_2 - y_3) + y_4)t^3 + \big(3(y_1 - 2y_2 + y_3)\big)t^2 + \big(-3(y_1 - y_2)\big)t^1 + y_1 t^0 , \qquad 0 < t < 1$$

A tool for keyframe-based timeline mechanism to store the state of the Bezier curve at each keyframe and then animate the pot object via animation interpolation was developed. Each keyframe stores the state of four 2D control points and an extra scale value for the size (scale) of the object. By using this tool and manual observation of the expert videos, an artist is able to reanimate the object in visual environment. In the future, this tool may also be used to automatically generate 3-D pottery object animations in real time for the virtual or real pottery games, by regularly adjusting the positions of the four control points, based on the hand gestures performed.



a) State at keyframe No:3 b) Interpolated state c) State at keyframe No:4

Figure 28: Pottery Object animation tool using keyframe-based timeline mechanism

4.3.3.2 Online object visualization

An online pot visualization approach is necessary when a student is seated to form the clay. To do so, the point cloud of the clay extracted from the depth camera should be used. To retrieve the related 3D points in the depth camera scene, a segmentation algorithm which operates in real-time is necessary. In Figure 29 shows

the result of such a segmentation algorithm (the black region belongs to the pottery object);

Figure 29: Black segmented region will be used to extract 3D point cloud of pottery object

When the segmentation algorithm marks the region of the clay (pottery object), the 3D positions are calculated automatically from the depth values of these 2D coordinated that belong to in clay segment of the depth image. Figure 30(a-b) demonstrates the point cloud for the pottery object;

a) Front view of pottery object point cloud

b) Isometric 3D view of pottery object point cloud

Figure 30: Point cloud extracted from segmented depth image

Since the pottery object is generalized with a 2D cubic polynomial, non-linear regression using least square method can be applied to the point-cloud data that belongs to the pottery object. To apply the data, the 3D point cloud data is projected to 2D cubic polynomial space and then the polynomial coefficients are calculated by least squares and the pottery object is extracted from the depth data. Figure 31 shows how the least squares algorithm is used to fit a curve to the point cloud data from the Kinect data in Figure 29 and the final visualization of the 3D pottery object;

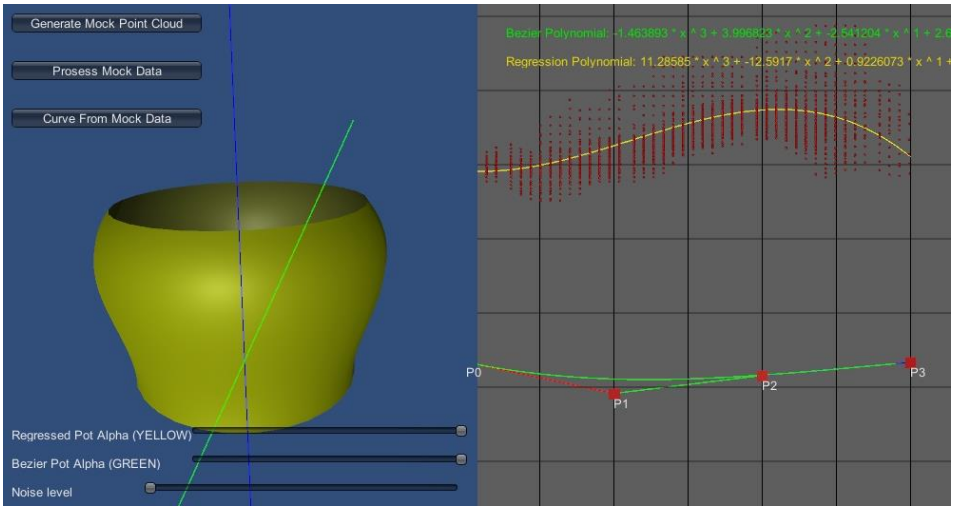


Figure 31: The yellow polynomial curve is regressed from the red dots which is the pottery point cloud projected to 2D polynomial space. As a result of the calculated cubic polynomial, yellow 3D pottery object is generated

4.3.4 Performance evaluation

The same algorithm that is used for gestural performance evaluation in contemporary music use case, described in 4.4.3.3, is also used for the evaluation of pottery gestures, based on machine learning techniques for the modelling and comparison of gestural performances. The gesture recognition accuracy, machine's ability to recognise learner's gestures, which is calculated after the execution of the gesture, based on the Precision and Recall metrics will be used as a score.

4.4 Contemporary Music Composition

4.4.1 Introduction

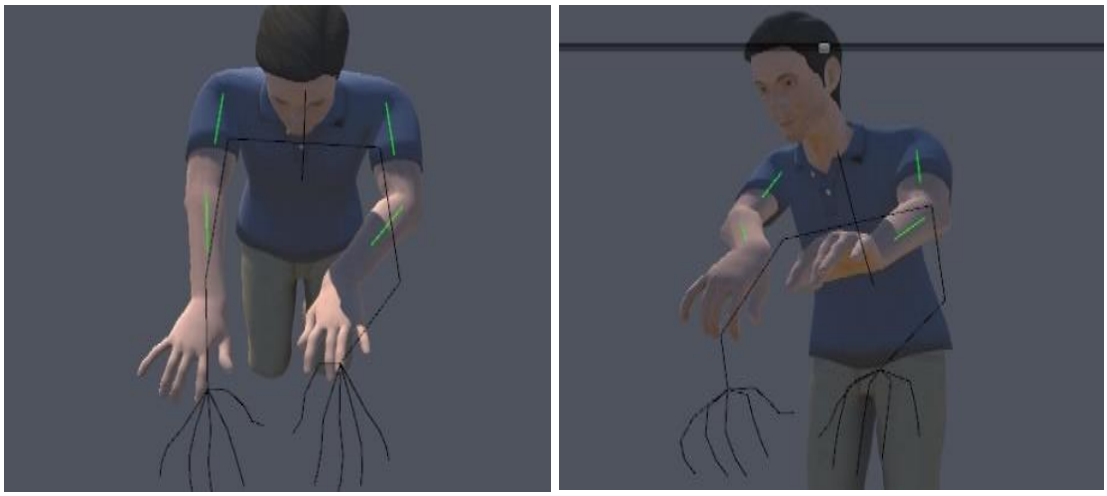
The Contemporary Music Composition game-like application aims to develop a novel intangible musical instrument (IMI), which maps natural gestures performed in a real-world environment to music/voice segments. Besides, the emotional status of the performer is also planned to be part of this multimodal human interface. The main objective of this game is to let the learner observe and practice the basics of new innovative intangible musical instrument by using various sensors and gaming interface.

During this third period, significant improvements were made regarding a) the avatar visualization, b) enhancements in the EmoActivity and c) implementation of a new "Gesture-Emotion activity with augmented musical score". These improvements are presented in the following sections.

4.4.2 Avatar Visualization

For the full body animation, Kinect and leap skeleton data are fused into 51 joint positions. Since character is in seated posture, only 9 upper body joints are used from Kinect device and 42 hand joints are used from leap device. Multiple Ad-hoc algorithms are implemented to extract the rotation of joints from joint positions. At final version of the implementation, all joint rotations are calculated. In the previous versions, IK (Inverse Kinematics) was used to animate the arm of the avatar. While, IK is successful in some arm postures, it could fail in others because of the nature of the approach. This is because IK calculated the arm posture from the end effector,

which are wrist positions in our case. Since infinitely many arm postures (configurations) may result with same end-effector position, all these different arm postures map to the same animation when IK is used. As a result, we decided to calculate arm joint rotations from positions as well. Figure 32 illustrates the avatar animation and its corresponding stick figure. The stick figure is directly drawn from the detected joint positions and is used for the evaluation of avatar animation.



a) Avatar posture from upper camera b) Avatar posture from side camera

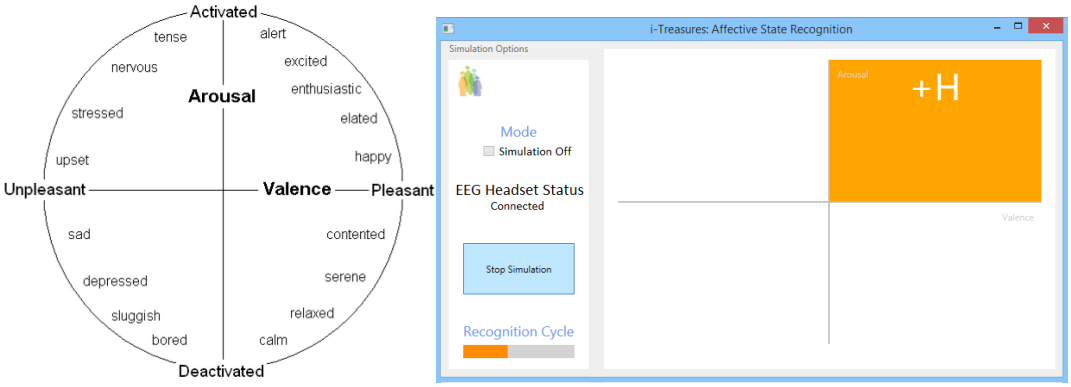
Figure 32: Avatar animation and the related stick figures

4.4.3 New and updated activities

4.4.3.1 Updated Emotion activity

EmoActivity is a game-like application that aims to help the user learn and handle certain affective states and transitions towards augmented artistic expression using the Intangible Musical Instrument. According to the game scenario, EmoActivity prompts the user to reach and sustain a series of specific affective states based on the level of the game and the current affective state of the user. The first version of EmoActivity was presented in deliverable D5.2 “First version of 3D visualization for sensorimotor learning”.

The affective states are represented by the 2D model of valence and arousal (Figure 33a). Valence denotes if an affective state is positive or negative, while arousal constitutes a measure of excitation. Four affective states are engaged in EmoActivity: (a) positive valence – high arousal, (b) positive valence – low arousal, (c) negative valence – high arousal, and (d) negative valence – low arousal. The affective state detection is based on electroencephalography (EEG) data analysed by the affective state recognition software (Figure 33) presented in deliverables D3.3 “Final report on ICH capture and analysis” and D3.4 “Final version of ICH capture and analysis modules”. EEG signal are acquired by the EPOC wireless recording headset (Emotiv Systems Inc., San Francisco, CA) (Figure 34a) that bears 14 channels, referenced to common mode sense (CMS) – driven right leg (DRL) ground (Figure 34b).



(a) (b)

Figure 33. (a) 2D valence-arousal space. (b) Affective state recognition software

(a) (b)

Figure 34. (a) EPOC wireless EEG headset by Emotiv. (b) Electrode positions of the EPOC headset according to 10/20 International System.

As far as the game scenario is concerned, EmoActivity consists of three levels:

- Level 1: The user is asked to reach a certain affective state (within 20 s) and sustain it (for 40 s). If the target affective state is reached, the user is awarded with: 5 points if the affective state is not sustained or 10 points if the affective state is sustained. In case the target affective state is not reached within 20 s the game stops and zero points are credited to the user.
- Level 2: The user is asked to reach a certain affective state within 60 s (phase 1) and then, if he/she succeeds, reach another affective state within the same period of time (phase 2). Ten points for each successful phase are awarded to the user.
- Level 3: In a similar way to Level 2, Level 3 requires the user to reach 3 consecutive affective states. In other words, Level 3 comprises three phases and the user continues to the next phase only if the current phase is completed correctly. Again, 10 points for each successful phase are awarded to the user.

Prior to the beginning of each level, the current affective state of the user is calculated (over a 20 s long period of time) and the initial target affective state is calculated based on this result.

In order to facilitate emotion elicitation, some sort of stimuli is presented to the user. The available options are:

- Affective images drawn from the International Affective Picture System (IAPS) database [2],
- Affective sounds drawn from the International Affective Digital Sounds (IADS) database [3],
- Affective videos drawn from the LIRIS Annotated Creative Commons Emotional Database (LIRIS-ACCEDE) [4],

The stimuli items, regardless of the type, are randomly presented to the user. The images are presented for 5 s each, while the sounds are 6 seconds long and the videos are 8-12 seconds long. When the affected sounds are engaged, the entire screen is painted to the colour that corresponds to the target affective state (see D5.2).

At the first version of EmoActivity, the user could only play Level 1 and the emotion elicitation was based on the affective images. Currently, the back-end implementation of the second and the third level is completed and the communication with the Unity 3D game engine is expected to be integrated soon. Moreover, the presentation of affective sounds and affective videos has been implemented but, currently, the selection of the type of stimuli is available through hard coding. A corresponding UI enabling the user to perform the selection intuitively is expected to be developed in the final implementation (to be provided in D5.7 and used in the demonstration and evaluation phases of the project).

4.4.3.2 Gesture-Emotion activity with augmented musical score

As it is mentioned in the previous deliverable D5.2, we have designed and developed a visualization module named "augmented music score", which is integrated into the Contemporary Music Composition game. The goal of this augmented music score is to facilitate the access to the knowledge of the expert, both gestural and emotional. The proposed "augmented music score" incorporates:

a) the music score from the composer,

b) gestural annotations (2D minimalistic representations of his/her upper body) of the most essential static (postures) and dynamic (gestures) phases of his/her movements for given music measures from the music score, and

c) emotional annotations describing the affective state of the expert performer while s/he performed given music measures from the music score. The affective states are four and they are defined by two levels of valence (positive or negative), which represents the perception of emotions as being either positive or negative, and two levels of arousal (low or high), which indicates the degree of intensity of an emotion. Four colours represent each one of the four emotional states, blue (positive valence – low arousal) refers to the general category of relaxation-related feelings, orange (positive valence – high arousal) to positive excitement-related feelings, grey (negative valence

– low arousal) to sadness-related emotions and red (negative valence – high arousal) to anger- and fear-related states, as in the EmoActivity game-like application.

Figure 35 presents an example of the augmented music score, which indicates the musical notes, the emotional and gestural annotations as well as the total time for each gesture and for each emotional status.

The concept of the augmented music score is the following. The composer can train the system with his/her musical gestures along with sounds and annotate the musical excerpt with emotional labels. Therefore, an association of the sound to the template gesture is created as well as each sound is linked with each template gesture (learning or training phase). Then, in the recognition phase or performing phase, the performer performs/imitates the same expert gesture with which the system has been trained. The basis of imitative synthesis is to make a gesture, which is representative enough to re-synthesize a plausible imitation of the original sound. Therefore, the gesture is being recognized in real-time, meaning the system estimates the gesture in real-time. As a result, the system provides to user, continuously output information about the gesture, which is probabilistic estimations (likelihoods).

Additionally, the performer can see the annotated emotional labels and the recognised levels of arousal and valence, which correspond to his/her current emotional status. The recognition of the emotional state takes place based on the affective state recognition algorithm that was developed within WP3 (see deliverable D3.3 "Final report on ICH capture and analysis). Here, the only difference is that the recognised valence level is estimated as the majority of support vector machines-based classification results within a varying time interval, as imposed by the music score, rather than within a four-second interval as reported in D3.3.

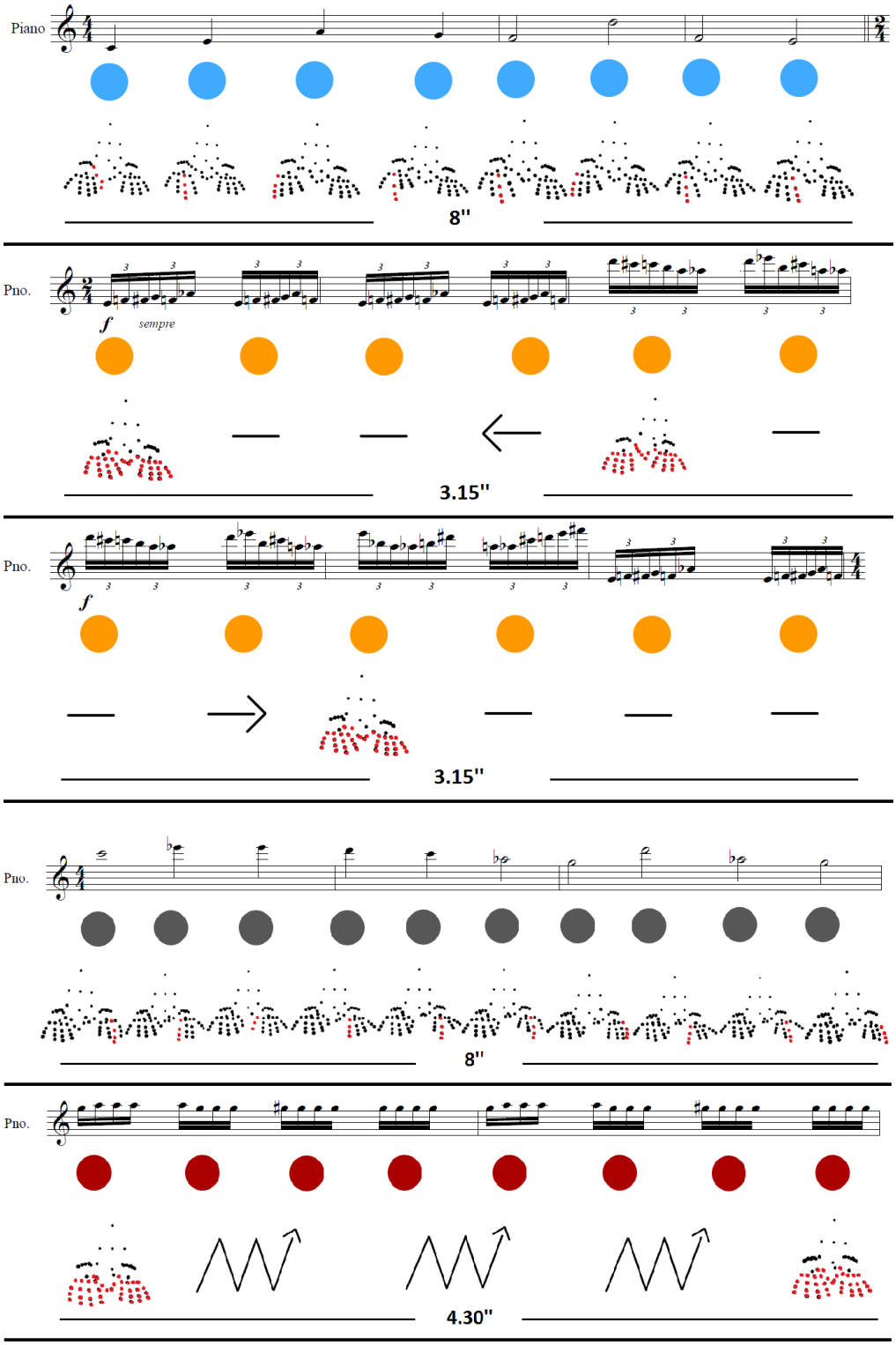


Figure 35. Augmented music score

According to recognised valence and arousal levels, the sound can be modulated nearly in real-time, i.e., every 1.5 seconds. More specifically, the level of arousal (high or low) modulates the loudness. A low level of arousal, which is mapped to the value of '0', produces a zero volume level-signal, while a high level, which is mapped to '1', produces a 75.6 dB volume level-sound. The loudness modulation occurs exponentially and with a delay of four seconds, in order the transition to be smoother and more fluid. Additionally, pitch is mapped to the level of valence. The technique used is pitch-shifting. The original sound has no pitch shifting. Therefore, according

to the levels of valence (positive or negative), the pitch of the re-synthesized sound is shifted linearly, by raising or lowering the pitch of the original sound (for more information please check D4.4).

Apart from the audio feedback that is given to the performer, the system, in the end of each gesture and empathy task, displays also visual feedback for the correct or not performed gesture and affective state reached with a green (correct) or red (wrong) annotation respectively (Figure 36).

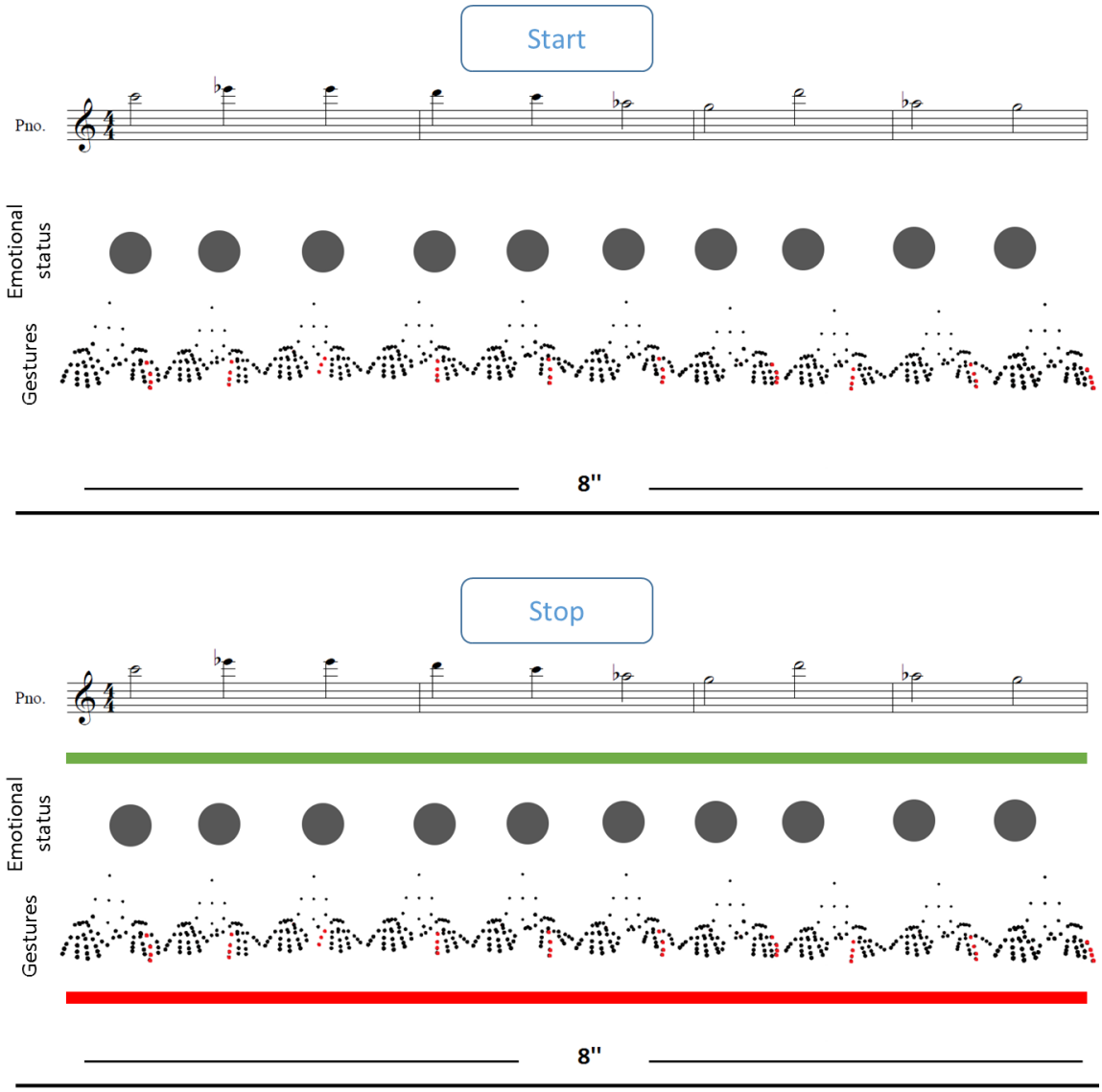


Figure 36. An example of the visual feedback to the user

A first implementation of the user interface for this activity is displayed in Figure 37.

Figure 37: The interface of the gesture-emotion activity with augmented musical score

4.4.3.3 Performance evaluation

The recognition engine, namely 'x2Gesture', has been developed in the case of Contemporary Music Composition. It implements machine learning algorithms, such as Particle Filtering and State Space techniques. It includes learning (or training) phase and performing (or recognition) phase (as it is already mentioned in the previous Section 4.4.3.2). x2Gesture is based on one-shot learning, meaning that only one sample of gesture is used to train the system. The performance evaluation that we have presented in D5.2, is integrated in the new recognition algorithm. More specifically, the evaluation method of the user/learner performance is being computed from the differences between the expert's and learner's performance. So the evaluation of the learner is based on the L1(absolute) distances between his/her gesture and the gesture of the expert in each axe:

$$Score\_x = \frac{\Sigma_{t=1}^{n}\left|(X_{RHAND}^{Learner} - X_{LHAND}^{Learner}) - (X_{RHAND}^{Expert} - X_{LHAND}^{Expert})\right|}{n} \quad (1)$$

$$Score\_y = \frac{\Sigma_{t=1}^{n}\left|(Y_{RHAND}^{Learner} - Y_{LHAND}^{Learner}) - (Y_{RHAND}^{Expert} - Y_{LHAND}^{Expert})\right|}{n} \quad (2)$$

$$Score\_z = \frac{\Sigma_{t=1}^{n}\left|(Z_{RHAND}^{Learner} - Z_{LHAND}^{Learner}) - (Z_{RHAND}^{Expert} - Z_{LHAND}^{Expert})\right|}{n} \quad (3)$$

where *n* is the number of the expert time stamps.

5. Novel Generic Framework for the creation of dance/body-motion based game-like applications

5.1 Introduction

Except from the aforementioned game-like applications dedicated to different ICH sub-use cases, our efforts also focused on the design and implementation of an xml-driven game development framework, which aims to provide a basis for those who want to create their own game-like applications for learning and training of dance and/or any other physical (body-motion-based) activity. As explained previously, this framework consists of two components: i) an interface that allows the user to design

the game scenario (define activities/ exercises, provide small descriptions, select performance evaluation algorithm, etc.), capture the necessary expert motion data and save this information in an xml file (this interface was developed by UMONS under Task 3.2), and ii) a customizable generic game, which can be automatically configured based on the XML output of the game design interface (this module was developed by TT under Task 5.2). The architecture of the generic game is similar to that of the Tsamiko/Walloon/Calus dance games. The information for the game activities and references to all necessary expert data for the "observe" mode are parsed from the xml file. In the "practice" mode, the aforementioned interface captures the learner's performance using off-the-shelve sensors like Kinect, compares this performance with the corresponding expert performance and, then, sends the evaluation result to the game-like application. This framework could be also used for creating educational game-like applications for a lot of other domains related to human motion, such as physical exercise, physiotherapy, rehabilitation, etc.

5.2 Module for game design

In order to design and develop serious game-like applications for dance learning and other activities involving full body gestures, the ITGD (i-Treasures Game Design) module has been developed by UMONS to facilitate this task. ITGD is a simple interface based on the MotionMachine framework (see D4.5 *"ICH Indexing by Stylistic Factors and Locality Variations")* that allows the user to design the game, record and annotate motion capture data and evaluate gestures. The overall architecture of this module is shown in Figure 38.

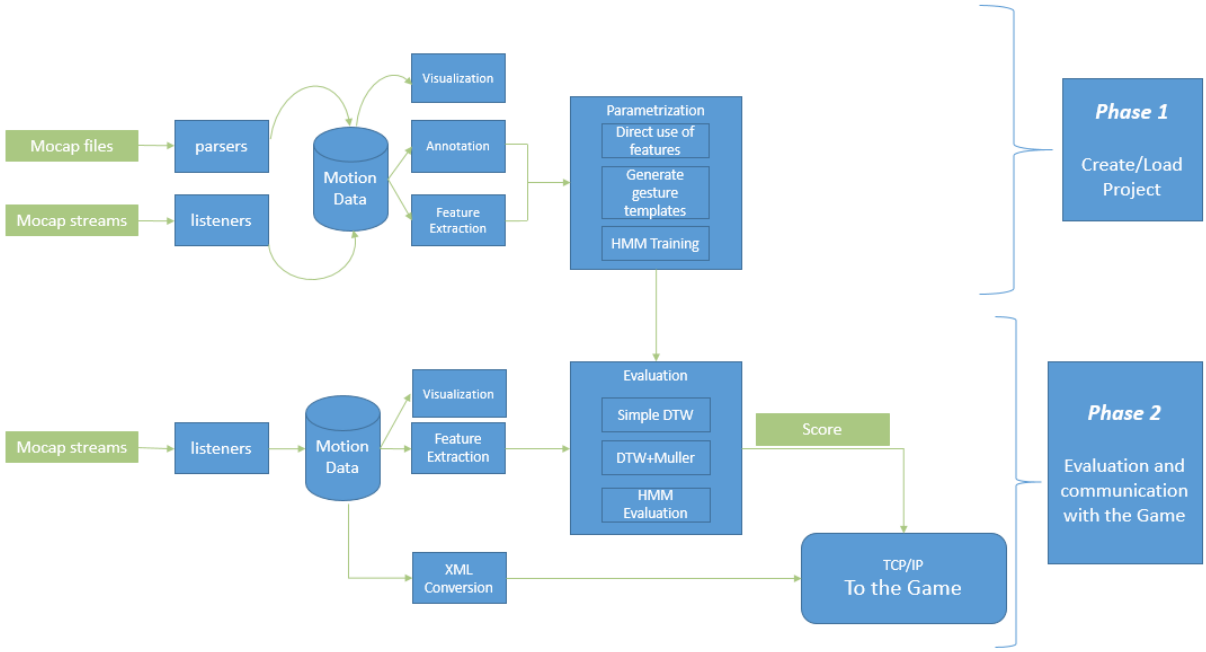


Figure 38: Overall architecture of the ITGD module.

As we see in Figure 38, the module is divided in two phases: design phase and runtime phase.

Phase 1: Designing the game

In this phase, we design the general architecture of the game. A screenshot of this phase of the project is shown in Figure 39. In this phase, we specify the name and the description of the project and also how many activities (levels) and the number of

exercises in each activity. Each exercise represents a gesture to be learned. The interface allows adding new exercises or editing existing exercises. In both cases, a new window appears (Figure 40) that allows editing the name and the description of the exercise, recording or loading mocap data and annotation of this data.

The interface allows also selecting the evaluation tool to be used. Until now, two algorithms are integrated that are DTW (where the sequence is warped to fit a reference sequence and a distance-like quantity is computed between these sequence in order to estimate a score and HMM evaluation described in details in the Section 2.4.2 of the deliverable D4.5.

Once the game has been fully designed, the project (mocap data and annotations) is saved and an “xml” file is generated. This “xml” file is used to set up the generic game (Section 5.3). The generated xml file contains the path to the project and the recorded data for each exercise in addition to the selected algorithm to be used for evaluation.

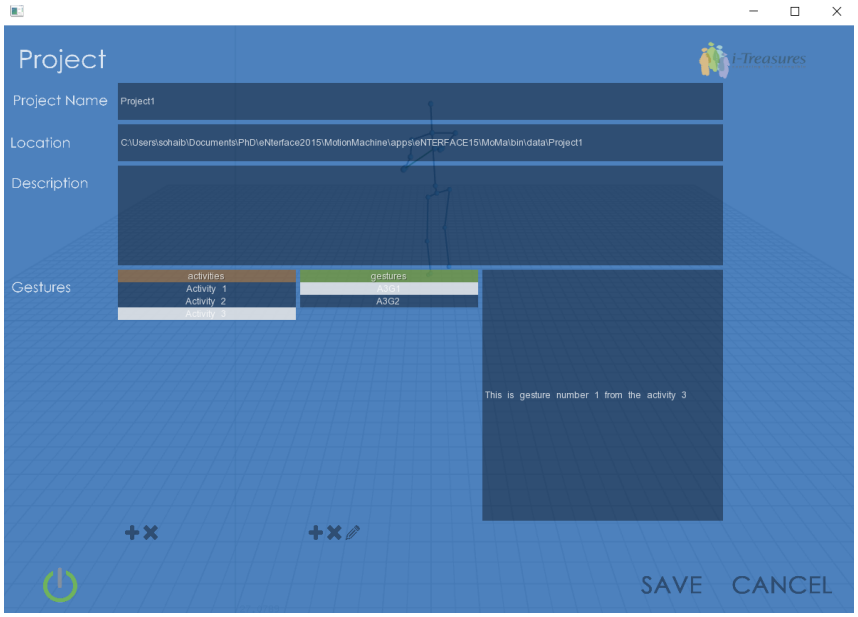


Figure 39: Screenshot of the project generation within the ITGD module.

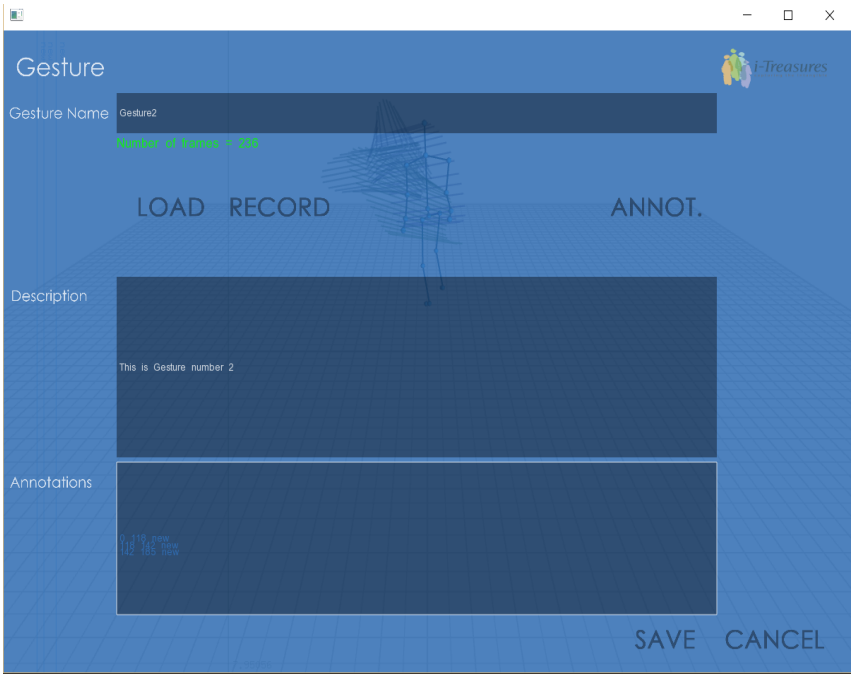


Figure 40: Screenshot of the gesture creation/edition within the ITGD module.

Phase 2: Runtime

In this phase, the module communicates directly with the generic game framework. It streams, in real-time, skeletal data received from a Kinect (or other device) to the game over TCP/IP and sends an overall score once the performance is finished. Each frame that is sent to the game contains positions, rotations and confidence scores of 25 joints in addition to timestamps.

5.3 Generic framework for dance/body-motion based game creation

The generic framework, developed by TT, can support all basic functionalities of a standard dance/body-motion-based game, such as those supported by the Tsamiko and Walloon games. Because this is a preliminary version of this framework, the game architecture is very close to the Walloon dance in terms of visualization. In the next version, the user should have the possibility to select the avatar and the environment.

How to:

Once the game is fully designed using the ITGD Module, described in Section 5.2, the files are copied to the generic game folder. The game will read the generated “xml” file and reconfigure itself in order to have the appropriate number of activities and exercises and also to display the correct name of the game. The expert avatar is animated in each exercise using the appropriate motion capture data where the path to this data is specified within the “xml” file. The game communicates with ITGD in runtime phase and shows the score of the performance to the student. If the performance if good, the game plays the next exercise, until the end of the game.

6. Game Analytics

In this section, we describe the general structure of game analytics which are metadata regarding the learner’s performance and activity, as he/she is playing or observing the games. These analytics are used to extract and evaluate results

regarding the applications and learner behaviours. This information will allow experts to discover meaningful patterns in learners' data, which, in turn, will allow creating more engaging and more instructive games. Analytics and web platform integration usually don't require much input from the users as this functionality is automatically handled by the game and the LMS server itself. The analytics data from each play session is collected by the analytics system and then this data is sent to LMS server for processing. Some of this processing occurs online and respond to user immediately and some of it will be saved on servers to assist researchers to make sense of the collected data and to be used on future researches. Performance and other analytics data is recorded to an XML file (Figure 41) and after the game session has ended, this file is sent to the LMS server. User authentication is performed by the login system. First, users need to sign up from the LMS web site. From here, they can select the games they want to play and gain access to functionalities they want. System grants permissions to these users. When users log in to game system with their username and password, they are able to play these games. Furthermore, sync feature is enabled so that every user's data is saved when they exit from the game and it is automatically loaded into the game when they are logged in.

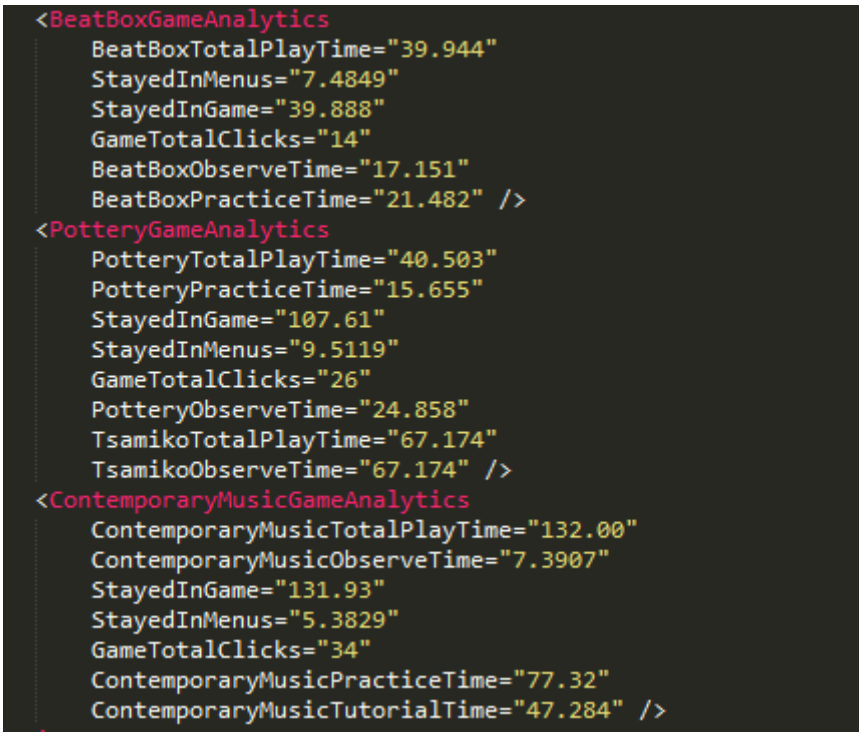


Figure 41. XML file with game Analytics

As seen, there are fields which show the time spent in menus, games (in observe and practice modes), as well as menu clicks and total clicks. This is quite important to see if there any part of the game which impacts the learners in any negative way. For example some particular feature of the game may discourage users to try advanced practice. Furthermore, if in some games certain features are enjoyable and encourage users to play and learn more, these best practices may be replicated in other games, if possible. Analytics XML also contains the language and guidance preferences of the users to be used in further development of the games. For instance, the game analytics may suggest implementing more detailed user guidance methods that will improve the learning curve and the educators' ability to understand students' learning performance.

7. Web Version of Game-Like Applications

The observe mode of the game-like applications developed within the scope of i-Treasures are also integrated with the web platform (http://i-treasures.multimedia.uom.gr/webplayer/WebBuild.html). To do this, we have made use of a web player plugin of Unity, which enables users to view 3D content created with Unity directly in most widely used browsers (Firefox, Safari, IE). We have included all the data necessary for the games before integrating them into the web platform.

8. Conclusions

This deliverable described the final version of the 3D visualization module developed as an outcome of the studies carried out in WP5. The 3D visualization for sensorimotor learning module introduced eight different game-like applications for the four ICH use cases. This deliverable mainly focused on providing minor/major updates regarding the game-like applications compared to status on deliverable D5.2.

The developed 3D visualization module supports people to learn or master different types of ICH using virtual tutors. Sensorimotor learning lets users start from any step or let them exercise as much as they want or till the virtual tutor tells them that they are ready for the next step.

In addition, a novel “generic framework for the creation of dance/body-motion-based games” was developed that allows experts to design and create their own dance games by just changing either built-in assets or in-house-captured motion data. We believe that this framework is an important contribution as it could be also used for creating educational game-like applications for many other domains related to human motion, such as physical exercise, physiotherapy, rehabilitation, etc.

Apart from this novel “generic” game framework, two additional important contributions of this Deliverable are:

- the design and development of a low-cost singing game (for Byzantine music), which can be extended in the future to support additional use cases in the future.
- an additional activity, which was added in the Contemporary Music Composition game, to visualize the augmented music score as well as the errors with respect to an expert performance, both regarding to gestures as well as emotions. This tool is expected to facilitate the access to the knowledge of the expert, both gestural and emotional.

The 3D visualization module will be further tested, optimized and integrated with other tools developed in WP5 under Task 5.5 System Integration. The final versions of the game-like application implementations will be delivered in Deliverable 5.7 – “Final Version of Integrated Platform”, before they are ready for use in the demonstration and evaluation phase of the project.

9. References

[1] A. Kitsikidis, K. Dimitropoulos, E. Yilmaz, S. Douka, N. Grammalidis, "A Game Like Application for Dance Learning Using a Natural Human Computer Interface", HCI International 2015, L.A. USA

[2] P. J. Lang, M. M. Bradley, and B. N. Cuthbert, “International affective picture system (IAPS): Affective ratings of pictures and instruction manual,” University of Florida, Gainesville, FL, Technical Report A-8, 2008.

[3] M. M. Bradley and P. J. Lang, “International affective digitized sounds (IADS): Stimuli, instruction manual and affective ratings”, The Center for Research in Psychophysiology, University of Florida, Gainesville, FL, Technical Report No. B-2, 1999.

[4] Y. Baveye, E. Dellandrea, C. Chamaret, and L. Chen, “LIRIS-ACCEDE: A Video Database for Affective Content Analysis,” IEEE Transactions on Affective Computing, 2015

[5] Mellish, L., ‘The Romanian Căluş Tradition and Its Changing Symbolism as It Travels from the Village to the Global Platform’, [Online] Available at: http://mainweb.hgo.se/ [13 July 2013], 2006.

10. Appendix: User manuals

10.1 Tsamiko

Tsamiko game is for learners who want to learn Tsamiko dance by observing experts' movements and by repeating the dance steps in front of a Kinect v1 or Kinect v2 camera. The game software guides the user through activities and teaches him/her how to dance Tsamiko properly.

After the main menu, the learner can select the Tsamiko dance in the Rare Dance Interactions menu. Here, the user selects "Tsamiko" from the sub use-case menu. Here the user can click the "Getting Started" button to start a tutorial of the game (Figure 42). This section not only gives information about the game (including the GUI of the Tsamiko game and the Kinect sensors used), but also about Tsamiko dance in general. A virtual tutor (often referred to as the "wise guy") appears and greets the user and explains the functions of the different panels in the game.

Figure 42: Virtual tutor

In the "Observe" mode (Figure 43), the learner is expected to watch the expert performing the Tsamiko dance. When starting this mode the virtual tutor appears and tells the learner in which mode he is. Apart from observing the expert performance, this part also allows users to get familiar with the different elements and functionalities of the game, such as the animation player and the game environment. In this mode, performances of the Tsamiko experts, captured by Kinect sensors are displayed. The performances are shown in the form of videos and by animating the 3D avatar with the respective movements of the expert dancer.

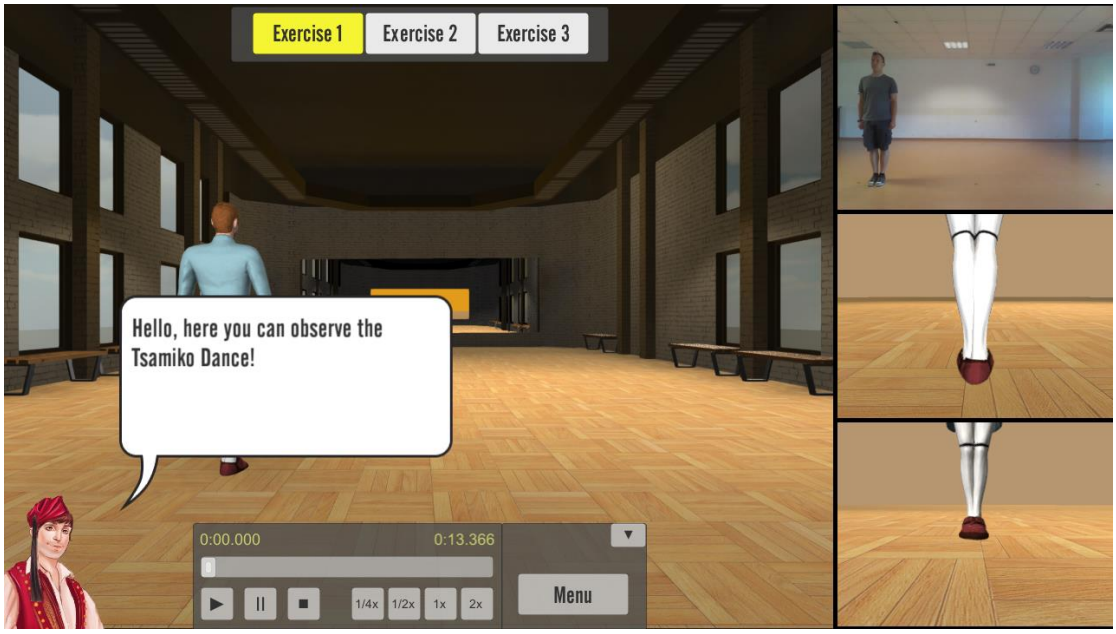


Figure 43: Screenshot of the Tsamiko observe mode.

Each expert performance is divided into different parts (activities and exercises) so that the user can learn the Tsamiko dance more effectively. Also, the user can interact with the free camera by pressing the Left-Alt key and moving the mouse and/or zoom in/out with the mouse wheel by scrolling.

In the “Practice” mode (Figure 44), the user is expected to have some familiarity with the game, thus the virtual tutor makes a brief appearance and the countdown starts. After the countdown reaches zero, the dance starts and the user is expected to imitate the experts’ movements. After the end of the learner performance, a score and feedback text, such as “Outstanding performance! You are ready for the next exercise/activity!”, or “Your movements need to be improved, pay attention to the expert’s movements”, is displayed in the wise guy speech balloon. In addition to the final score, displayed to the user after the end of each exercise, an instant score is also displayed during the user performance, which constantly measures the correctness of the performance during a specific time segment. With this feedback, the user can understand whether his/her moves are correct or if they require any alteration. In addition, a colour coded scale is provided. After the user reaches a certain performance threshold, the learner can proceed to the next exercise. When all exercises are completed, the final Challenge is unlocked.

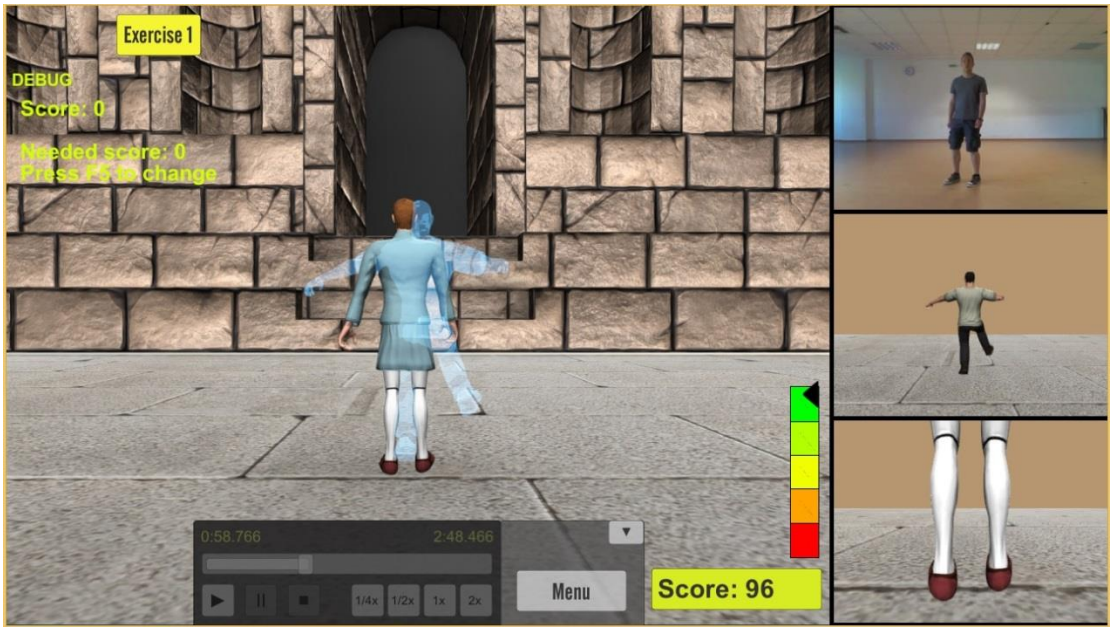


Figure 44: Screenshot of the Tsamiko practice mode.

For the evaluating the learners' movements, the game module exchanges data with the WP3 capture and evaluation module, developed by CERTH.

10.2 Calus

The Calus game is very similar in to Tsamiko game regarding both the game structure and the user performance evaluation, but different assets (3-D avatar and virutal tutor) and dance performances are used. The user can learn to dance Calus dance by observing the expert performance that is recorded by a Kinect v1 or Kinect v2 sensors and then practicing the same performance. The expert performance is divided into different parts (activities and exercises), so that the user can observe and practice each individual part. The game guides the user through each exercise.

In the main the menu, the learner can select "Calus dance" in a sub use-case menu displayed in the Rare Dance Interactions menu. The user can press the "Getting Started" button to start a tutorial of the game. This section gives general information about Calus dance as well as information about the game (including the GUI of the Calus game and Kinect sensors supported). A virtual tutor (often referred to as the "wise guy") appears and greets the user and explains the functions of the different panels in the game. This allows the learner to get familiar with the several elements of the game such as the animation player and the environment.

In the "Observe" mode of each activity (Figure 45), the learner is expected to watch the expert performing the Calus dance. When starting this mode the virtual tutor appears and tells which mode the learner is watching. Apart from observing the performance of the Calus expert, this part also allows users to get familiar with different functionalities of the game.

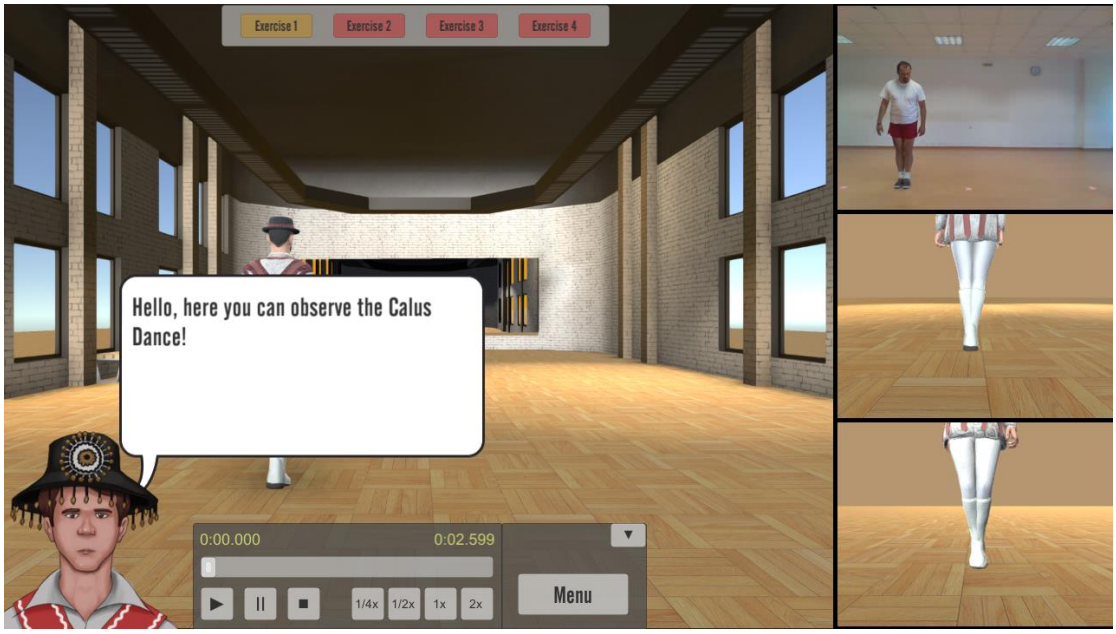


Figure 45: Screenshot of the Calus observe mode.

A video from the expert performance is shown and the Calus avatar is animated with the corresponding movements. Also, the user can interact with the free camera by pressing the Left-Alt key and moving the mouse and/or zoom in / zoom out with the mouse wheel by scrolling.

In Practice mode (Figure 46), the user is expected to have some familiarity with the game, thus the virtual tutor makes a brief appearance and the countdown starts. After the countdown reaches zero, the dance starts and the user is expected to imitate the experts' movements. After the learner repeats the particular exercise, a score and a feedback text such as "Outstanding performance! You are ready for the next exercise/activity! ", or "Your movements need to be improved, pay attention to the expert's movements " is displayed by the wise guy speech balloon. After the user reaches a certain performance threshold, the learner can proceed to next exercise. As in Tsamiko game, the instant score is also displayed during the performance both as in a numerical form as well as a color-coded scale.

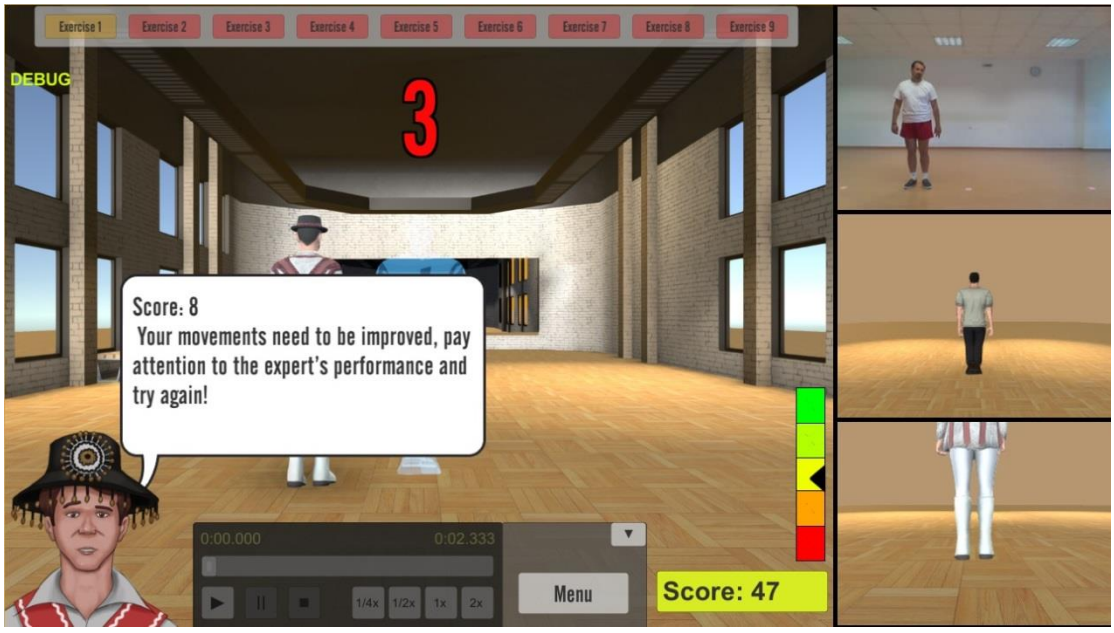


Figure 46: Screenshot of the Calus practice mode with user performance evaluation

For the evaluating the learners' movements, the game module exchanges data with the WP3 capture and evaluation module, developed by CERTH.

10.3 Walloon

The Walloon dance game has some notable differences with respect to the Tsamiko and Calus games. In this game, the expert's movements are captured by a high precision motion capture system, but as this system is very expensive, an inexpensive sensor, such as Kinect v2, is used for playing the game. Experts' video is not captured by the motion capture system, so currently the "Observe" and "Practice" screens don't display the experts' video.

Figure 47: Screenshot of the Walloon observe mode.

In the "Observe" mode (Figure 47), the learner is expected to watch the expert performing the Walloon dance. When starting this mode the virtual tutor appears and explains the mode the learner is watching. Apart from observing the performance of the Walloon expert, as captured by the motion capture system, this part also allows users to get familiar with different functionalities of the game. This allows the learner to get familiar with the several elements of the game such as the animation player and the environment. In this mode, performances of the Walloon experts are displayed. Furthermore, the 3D environment is a reconstruction of a Walloon village which users can explore.

In the "Practice" mode (Figure 48), the user is expected to have some familiarity with the game, thus the virtual tutor makes a brief appearance and the countdown starts. After the countdown reaches zero, the dance starts and the user is expected to imitate the experts' movements. After the learner repeats a particular exercise, a score and a feedback text such as "Outstanding performance! You are ready for the next exercise/activity!", or "Your movements need to be improved, pay attention to the expert's movement " is displayed by the wise guy speech balloon. A percentage score to the user is given. This can be interpreted by the user. A good performance is higher than 75%, a medium performance is between 75% and 50% and a bad

performance is below 50%. In the Walloon dance, errors usually occur because learners don't cross their feet and they don't bend their knees enough according to experts. After the user reaches a certain performance threshold, which is decided by the Walloon experts, the learner can proceed to the next exercise. After all the exercises are completed in an activity, that activity is concluded and learner is ready for the next exercise.

Figure 48: Screenshot of the Walloon practice mode.

For evaluating the learners' movements, the game module exchanges data with the Walloon game capture and evaluation module, developed by UMONS.

10.4 Generic Dance game

The Generic Dance Game Application is designed and developed to simulate variety of different dance training scenarios. It is a base game application to create many other dance teaching games. Like in other dancing games (i.e. Tsamiko, Calus and Walloon), the game will consist of any number of "Activities" which each contains any number of "Gestures" for users to practice. The game is customized by the ITGD Module for game design developed by UMONS (described in Section 5.2). The tool allow user to change the name of the game, change its number of "Activities" and their "Gestures". Users also can load from a file (UMONS tool supports different formats, c3d, txt, xml, v3d…) or record a new "Gesture" for the Expert movement.

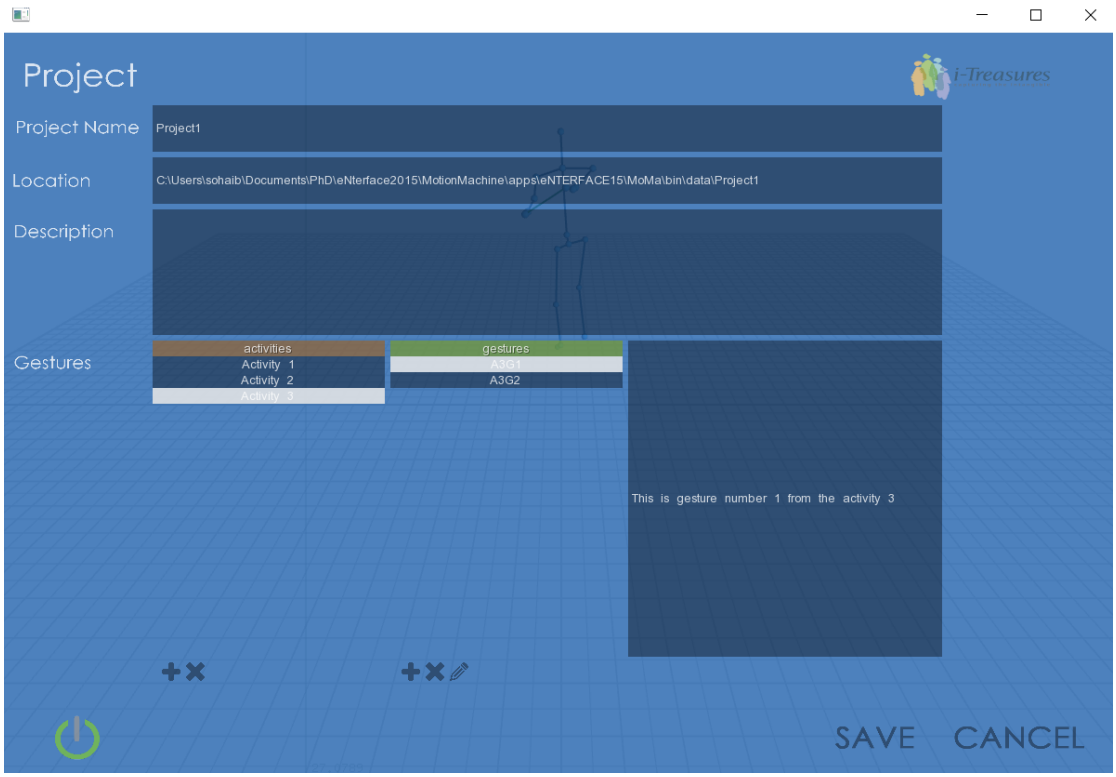


Figure 49: Customization tool main menu.

In the main menu users starts with creating a new project. There users can change the project name, its description, add or remove activities and add gestures to those activities.

Figure 50: Gesture menu.

The Gesture is the capture data of the exercise the learner will undertake in the game. Users can load it directly from a file (several different file formats are supported such as c3d, txt, xml, v3d) or capture the data in the tool themselves.

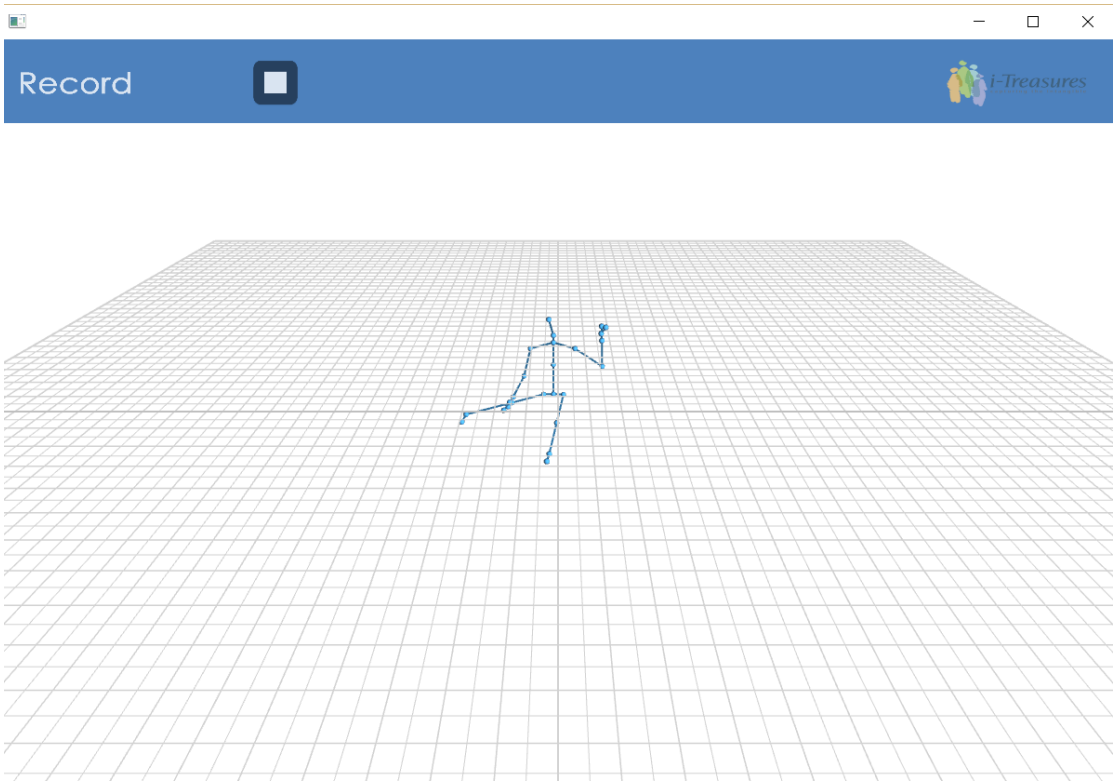


Figure 51: Record menu.

After all activities prepared and all gestures recorded/loaded then project users can save the project. After saving project.xml files, Data and Sequences folders will be created. Copying those files into the game folder will change the game.

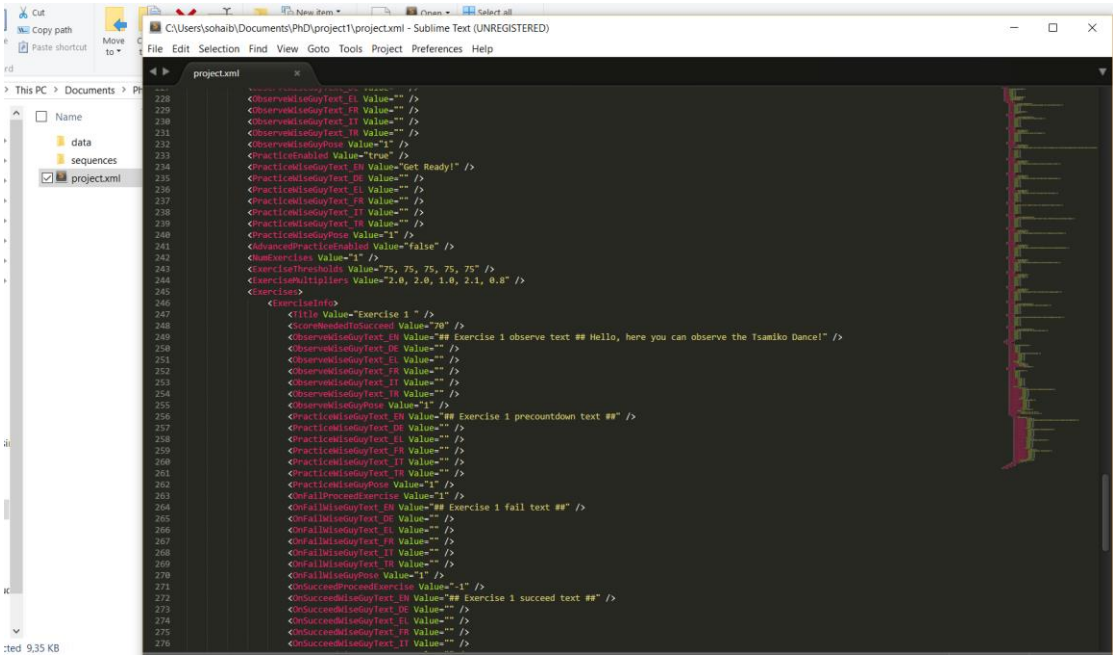


Figure 52: Game configuration (.xml) file.

Once the game is fully designed using the ITGD Module, described in Section 5.2, the files are copied to the generic game folder. The game will read the generated

“xml” file and reconfigure itself in order to have the appropriate number of activities and exercises and also to display the correct name of the game. The expert avatar is animated in each exercise using the appropriate motion capture data where the path to this data is specified within the “xml” file. The game communicates with ITGD in runtime phase and shows the score of the performance to the student. If the performance if good, the game plays the next exercise, until the end of the game.