Deliverable 5.3 - Gesture and Action Recognition

Release Date	30.11.2019
Version	0.1
Dissemination Level	Confidential

Project Number	822336
Project Acronym	Mingei
Project Title	Representation and Preservation of Heritage Crafts

Deliverable Number	D5.3
Deliverable Title	Gesture and Action Recognition
Deliverable Type	Report
Dissemination Level	Confidential
Contractual Delivery Date	M12
Actual Delivery Date	31/05/2020
Work Package	WP5 – DEVELOP & EXECUTE: Build Toolkit and Platform
Authors	Gavriela Senteri, Alina Glushkova, Vitto Nitti, Emmanouil Zidianakis, Xenophon Zabulis, Nikolaos Partarakis
Number of pages (incl. cover)	46

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 822336.
http://www.mingei-project.eu/

Executive summary

This deliverable describes the necessary scientific and technical work done for the gesture and action modelling, recognition and comparison, in order to build the interaction mechanism on which the embodied user experience in the museum will be based. To make the user able to interact with a machine (an installation in our case) it is necessary first of all a) to detect his body (input from 5.2), then b) to model and recognise his/her gestures c) compare them with a reference model. The comparison will trigger an interaction mechanism that may use different modalities (visual, auditory etc.).

The scientific and technical aspects of the 2 multimodal interactive techniques that will be used are described in this document.

In order to achieve the gesture modelling we introduce the Gesture Operational Model which describes the relationships between different body joints while doing a movement. More precisely it mathematically defines how gestures are performed based on assumptions that focus on the dynamic association of body entities, their synergies, and their serial and non-serial mediations, as well as, their transitioning over time from one state to another. Then, the assumptions of the Gesture Operational Model are translated into a simultaneous equations system for each body entity through State-Space modelling. The coefficients of the equation are computed using the Maximum Likelihood Estimation method. In the recognition phase, State-Space modelling is combined with continuous Hidden Markov Models to boost the recognition accuracy when the likelihoods are not confident. The performance of the algorithm (hybrid HMMs) has been evaluated using a glassblowing dataset that contains gestures from the routine of the creation of a glass carafe. The results are compared to the performance of HMMs, without the contribution of the State-Space representation method. The presented methodology surpasses the results of the HMMs and the results appear to be very satisfying and promising. Those recognition results are exploited in a preliminary stage, by mapping the gestures performed by the user of the cultural installation. Explicit mapping is used as a method for the sonification of the users' gestures. According to how well each gesture is performed, the pitch and panning of the produced sound is affected. These sound modalities work as instructions for the user to perform the gestures as close to the original ones as possible.

Furthermore, in this deliverable an alternative framework for natural, gestures based interaction is presented. The implementation of the infrastructure stems from the requirements of the Mixed Reality Surface where the users should not only interact with physical objects (through sonification) but also have a way of interacting with the UI of the system in order to execute specific workflow commands such as start, stop, pause, resume, etc. These requirements pose extra difficulties for the runtime recognition engine which is the main research part of this deliverable. As runtime recognition and sonification is a complex Computer Vision problem it was decided to take some of the burden of the recognition to an existing software platform which is the Microsoft Kinect SDK. This SDK in order to support seamless interaction regardless of the user skeleton size, rotation and distance from the screen needs calibration which is a process that in most of the cases results to reduced user experience. To cope with this situation Mingei facilitates through Nibbler a technique based on DTW (Dynamic Time Warping) that permits to align in time,

timeseries of different length and thus minimise the need for calibration as the gesture and posture recognition system dynamically adapt to the skeleton characteristics of each user but also to the speed of interaction seamlessly identifying gestures and postures.

The deliverable is structured as follows:

Section 1 introduces the context and purpose of this deliverable, specifically regarding the two alternative multimodal interaction technologies developed by the project. The first is targeting gesture modelling/recognition/comparison using the sound modality for interaction since user's movement is sonified based on the differences between the reference and the user's model. The second regards a gesture and posture recognition framework capable of running simultaneously with the sonification modality, allowing natural interaction with the UI of a Mixed Reality application.

Section 2 provides an overview of the state of the art regarding research efforts on movement modelling and the application of Machine Learning (ML) technologies for gestures recognition. Then it moves further to the elaboration of background work relevant to the field of movement sonification and the previous expertise of the consortium in the research subjects in question. Mingei's aim in this field of research is to get advantage of the existing knowledge, combine ML algorithms to statistical methods to give **the best results for real time gesture recognition** with cutting edge technology and the **creation of an installation for a cultural institution and a chance for a full cultural experience for the visitor**.

Section 3 provides a description of the methodology followed by Mingei for gesture modelling recognition and comparison for movement sonification. In this context, the methodology pipeline is discussed and each step is analysed in detail. Motion capturing took place in a glassblowing workshop at Nancy in France. Features were extracted from those data, which were then represented in a State-Space form and were used as an input for training the proposed hybrid HMMs algorithm. Whenever the HMMs appeared not to be confident concerning gesture recognition, the forecasting signal extracted from the State-Space representation and the use of ML estimation method was exploited to boost the recognition results.

Section 4 presents the infrastructure to be used for interacting with the UI of the final system while sonification will be employed for interaction with physical objects. For the validation of the infrastructure in the context of a similar to Mingei test scenario, where the user is request to assume certain gestures, an existing software product of FORTH was used called the mimesis game. In this game the infrastructure was used to detect postures in order to measure recognition accuracy. This validation is not related with the final user based evaluation of the infrastructure that will happen in real life conditions and using the final Mixed Reality Surface in the context of the Mingei pilots' realisation.

Section 5 provides experimental results and future work.

This deliverable is submitted in the context of T5.3 of Mingei. This is the first version of the deliverable reporting progress achieved during the first year of the project. The second version of

the deliverable will be submitted on M24.

Keywords

Gesture recognition, movement modelling and representation, movement sonification, action modelling, gesture-based interaction

Document History

Date	Version	Author/Editor	Affiliation	Comment
05.22.2020	0.1	Gavriela Senteri	ARMINES	Initial draft
31.10.2019	0.3	Emmanouil Zidianakis, Xenophon Zabulis, Nikolaos Partarakis	FORTH	Updated draft – integration of multimodal interaction techniques
20.11.2019	0.3	Vitto Nitti	IMA	Revised version
28.11.2019	0.4	Margherita Antona	FORTH	Final QA review
05.05.2020	R1	Gavriela Senteri	ARMINES	Updated draft based on comments received by project monitors
15.05.2020	R1.1	Emmanouil Zidianakis, Xenophon Zabulis, Nikolaos Partarakis	FORTH	Updated draft based on comments received by project monitors
26.05.2020	R1.2	Gavriela Senteri , Emmanouil Zidianakis, Xenophon Zabulis, Nikolaos Partarakis	ARMINES, FORTH	Final draft after review

Abbreviations

HMMs	Hidden Markov Models
SS	State Space
SE	Signal Error
DTW	Dynamic Time Warping
FNN	Feedforward Neural Network
SVM	Support Vector Machine
ML	Machine Learning

Table of Contents

List of Figures

List of Tables

1. Introduction

This deliverable describes the necessary scientific and technical work done for the gesture and action modelling, recognition and comparison in order to build the interaction mechanism on which will be based the embodied user experience in the museum. To make the user able to interact with a machine (an installation in our case) it is necessary to first of all detect his body, using an input from Task 5.2. The next steps for an efficient interaction to be achieved are the recognition of the users' gestures and their comparison with a reference model. This comparison will finally trigger an interaction mechanism using different modalities (visual, auditory, etc.) that will work as a feedback or instructions for the user. These instructions will motivate the user to complete specific tasks as instructed from the installation itself.

We describe here 2 different multimodal techniques:

- The one is focusing on gesture modelling/recognition/comparison based on the State Space and Hidden Markov Models. The modality used for interaction is sound since user's movement is sonified based on the differences between the reference and the user's model. Sonification is the use of non-speech audio to convey information or perceptualize data[1]. It complements visualization in a way that helps the user to learn and interact to the installation.
- The other is based on DTW that permits to align in time, timeseries of different length and use mostly the visual modality since the user interacts with an avatar projected on a screen.

Both methods will run in conjunction in real-time to permit, complementing each other. The main concept is to provide two different systems one for detecting gestures related to the User Interface of the MR surface and another one related to the recognition of craft gestures. The distinction between these modalities will allow the system to depend on different sub-system for each task thus switching between modalities to reduce error rates, improve effectiveness and enhance user experience.

[1] https://en.wikipedia.org/wiki/Sonification

2.1 State of the Art

Movement can be defined as the change of someone's' position, while gesture is a form of non-verbal communication in which visible bodily actions communicate particular messages. To reach to the point of the movements' interpretation, thus, to gesture recognition, it is essential to understand the existing relationships among human body parts.

2.1. Movement modelling and representation

Each body part is strongly connected and affected by the movement of others. We only need to think of the movement of a person running to understand that some body parts need to work cooperatively with others for a movement to be achieved. Duprey, Naaim, Moissenet, Begon and Cheze [1] tried to explain the anatomy of the human body, mostly for clinical and ergonomic uses. Concerning mainly the use of statistical methods for mathematical movement representation, estimation and forecasting, Zalmai, Kaeslin, Bruderer, Neff and Loeliger [2], used linear state space models and provided an algorithm based on local likelihood for reliably detecting and inferring the gesture causing magnetic field variations, while Lech and Kostek [3] presented a system based on camera and multimedia projector enabling a user to control computer applications by dynamic hand gestures. Hand position tracking was achieved using Kalman filters and gesture recognition was supported by fuzzy rules.

2.2. Machine learning for gesture recognition

Movement modelling and representation methods lead to gesture estimation but don't allow the modelling of precise movement patterns and consequently their recognition, as well as taking into consideration qualitative aspects of human movement such as expressivity. These limitations can be surpassed with the use of machine learning methods. Several studies have been done in the past years in the field of gesture and movement recognition with the use of machine learning methods. Pedersoli, Benini, Adami and Leonardi [4], used a Kinect connected to an algorithm of hand-pose recognition of American Sign Language that uses HMMs, to achieve real-time recognition of static hand-poses and dynamic hand-gestures. Aggarwal and Cai [5] are within the few that modeled the body, to proceed to gesture analysis and recognition with tracking cameras, while Sideridis, Zacharakis, Tzagkarakis and Papadopouli [6], created a gesture recognition system for IMUs that uses the methods of FNNs and SVMs for gesture recognition. Yang and Sarkar [7] have used an extension of HMMs for gesture recognition using fragmented observations for their case.

Machine learning algorithms, such as those based on HMMs [13], Dynamic Time Warping (DTW) [14], Hierarchical Hidden Markov Models (H-HMMs) [15], Sequential Monte Carlo techniques [16] etc., are widely used for gesture recognition systems for continuous interaction. [17][18][19] successively developed a system based on a hybrid model between HMMs and DTW, called Gesture

Follower (GF), for both continuous gesture recognition and following, between the template or reference gesture, and the input or performed gesture (template-based method). It can learn a gesture from a single example (one-shot learning) by associating each template gesture to a 'state' of a hidden Markov chain [20]. During the performance, a continuous estimation of parameters is calculated in real-time, by providing information for the temporal position of the performed gesture. Time alignment occurs between the template and the performed gesture, as well as offering an estimation of the time progression within the template in real-time.

There are also some examples where real-time gesture recognition is performed, using again methods like Hierarchical HMMs or DTW [8] [9] [10], with some limitations though. Few of those limitations are the use of a 1D sensor, or the use of not automated methods -using statistical packages-for the case of training the system. However, in some other cases, HMMs have also been used in other similar cases [11] for virtual reality installations, where gesture is incorporated. Another limitation of HMMs is that observations are produced at the frame level, and as a consequence they do not support the transitions between segments [21]. Therefore, [14] [22] developed a system based on Hierarchical Hidden Markov Models (H-HMMs) with two levels for real-time gesture segmentation and recognition. Similarly to GF, it adopts a template-based method and implements one-shot learning. The system is trained with a single pre-segmented gesture, which is annotated by the user. Each segment is associated with a high-level state (segment state), which generates the sub-models of the signal level (lower level), encoding the temporal evolution of the segment [22] [23].

2.3. Movement sonification

Gesture recognition output can be used for real-time human-machine interaction, where different modalities are mapped to motion parameters to augment human movement in different contexts (learning, artistic performances, rehabilitation, etc.). The motion parameters values and/or gesture recognition output can be thus used to control the sound characteristics. The term sonification is used when variables are mapped to sound parameters by a function [28]; for example, each change in velocity results in a defined change of the sound's pitch or frequency. To sonify human movement, different sound to gesture mapping techniques can be used such as the explicit and the implicit [29] [30] ones.

2.3.1. Explicit sound to gesture mapping

In explicit mapping, gesture parameters are directly mapped to sound control parameters. According to [27], the method is efficient but the expressive power and consistency of direct relationships can be limited when the sound synthesis parameters are not perceptually meaningful. Explicit sonification was used by Volioti at al. where the users were invited to interact with an « Intangible Musical Instrument" by performing gestures with their whole body, moving their hands and torso [31]**.** Simple sound parameters (speed of sound) were connected to motion parameters

(acceleration of motion) to make the explicit mapping easily understandable by the user. However, this direct relationship creates limitations to user's expressivity as any change in the corresponding motion parameter affects the sound synthesis result.

2.3.2. Implicit sound to gesture mapping

To overcome some of the limitations of the explicit mapping an alternative solution has been defined: the implicit or indirect mapping that uses an intermediate model to express complex relations between motion and sound. Volioti et al. [21] during the i-treasures project (Intangible Treasures - Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures FP7-ICT-2011-9-600676-i-Treasures) combined statistical methods like State Space (SS) models to perform model estimation and forecasting. The forecasting error was computed defining the confidence intervals that are used as threshold within which a gesture is accepted as a correct one, along with Particle filtering for gesture recognition. This combination led to satisfying recognition accuracy results, improving what was previously done by Caramiaux [16] as well as earlier tries [21] [17] where no confidence intervals were used and HMMs or Hierarchical HMMs were used instead of Particle Filtering.

In both the implicit and the explicit sound to gesture mapping, sonification gives a motivation to the user to complete all his tasks/gestures by also reaching to a musical goal. Following the tempo of the sounds is also a helpful feedback on how well the gestures have been performed and in which way they need to be improved.

2.4. Previous connected work of the Consortium

During the Intangible Treasures - Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures FP7-ICT-2011-9-600676-i-Treasures, statistical methods like State Space model representation and maximum likelihood estimation to perform model estimation and forecasting, were combined. This gives a forecasting error that defines the confidence intervals as threshold within which a gesture is accepted as the correct one, along with Particle filtering for gesture recognition. This combination led to satisfying recognition accuracy results, improving what was previously done by Caramiaux, Montecchio, Tanaka and Bevilacqua[23], as well as earlier tries[24][17], where no confidence intervals were used and HMMs or Hierarchical HMMs were used instead of Particle Filtering. The aforementioned methodologies and research approaches do answer the question of what/which gesture is performed, but not how the gesture is performed from the expressive point of view. During the Intangible Treasures - Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures FP7-ICT-2011-9-600676-i-Treasures, an intangible musical instrument was created that incorporated the methods mentioned above.

Through the State of the Art, the proposed methodologies were either not corresponding to real time gesture recognition, the authors used for their experiments sensors with limitations themselves, like 1D sensors, or ready statistical packages (like SPSS, Eviews, etc.) were used, that didn't allow the authors to interact and experiment with them in order to improve the results to their best possible. The aim now is to get advantage of the existing knowledge, combine machine learning algorithms to statistical methods to give the best results for real time gesture recognition with cutting edge technology and the creation of an installation for a cultural institution and a chance for a full cultural experience for the visitor.

3. Methodology

For managing the creation of the cultural installation mentioned above, a methodology was created, as shown in Figure 1, consisting of seven different steps.

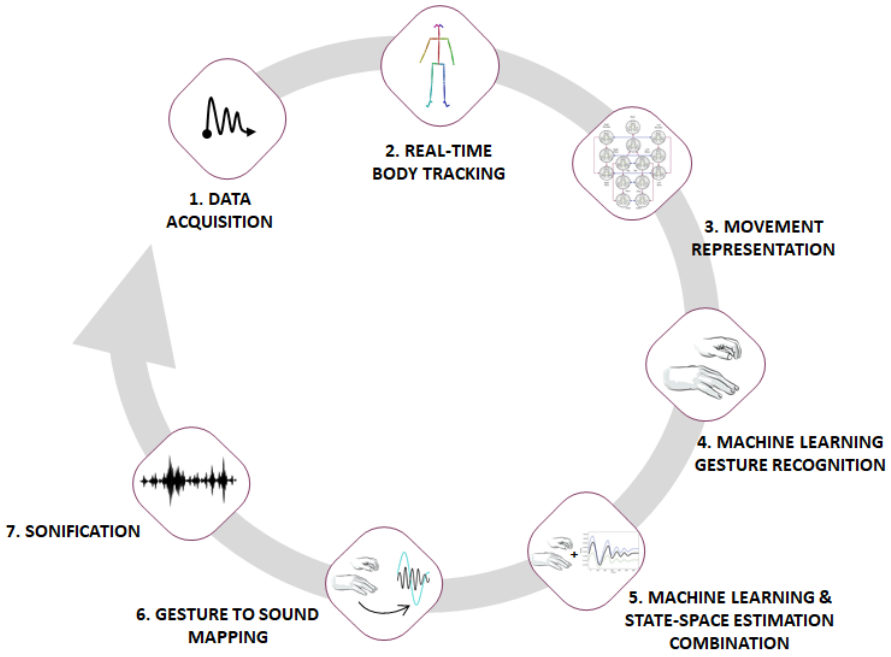

Figure 1: Methodology pipeline [34]

3.1. Data acquisition

The performance of the algorithm is evaluated with a dataset recorded in the CERFAV laboratory, at Nancy, in France. A gesture vocabulary has been defined in order to segment the whole glassblowing procedure into small human motion units.

The gestural vocabulary contains four gestures performed by a glassblower when creating a water carafe. The craftsman executes the gestures in a very limited space that is defined by a specific metallic construction. The craftsman puts the pipe on the metallic structure, to perform various manipulations of the glass using tools, such as pliers. More precisely, he starts by shaping the neck of the carafe with the use of pliers, then he tightens the neck to define the transition between the neck and the curved vessel, he holds in his right hand a specific paper and shapes the curves of the blown part and finalizes the object and fixes the details by using a metallic stick. In general, the right hand is manipulating the tools while the left is holding and controlling the pipe (Table 1).

Table 1. Glassblowing gestural vocabulary [35]

G_1: Fix details with pliers	G_2: Tighten base of glass	G_3: Make shape with paper	G_4: Fix shape

3.2. Real-time body tracking

After motion capturing and data recording, each image sequence of the dataset is imported to the OpenPose framework[2], which detects body keypoints (or joints) on the RGB image and extracts a skeletal model together with the 2D positions of each body joint [33] . These joints are not necessarily physical joints. They are keypoints on the RGB image which, in most cases, correspond to physical joint centres. OpenPose uses the neck as the root keypoint to compute all the other body keypoints. Thus, the motion data are normalized by using the neck as the reference keypoint. In addition to this, the coordinates of each joint are derived by the width and height of the camera. The extracted features for each joint, were the X and Y positions, as they are provided by OpenPose. More specifically, for the current dataset, the 2D positions of the head, neck and shoulder, elbows, wrists and hands were extracted, as they were proven to give optimal recognition results.

3.3. Movement representation

3.3.1. Gesture operational modelling and representation

Movement is a complex phenomenon and it needs to be studied both spatially and temporally. In order to model a movement, human movements need to be described in a way that will involve all the elements they depend on.

[2] https://github.com/CMU-Perceptual-Computing-Lab/openpose

Figure 2. Gestural dependencies, a glassblower, a silk weaver [36]

Observing the movements of a potter, a glassblower or a silk weaver we easily notice some existing dependencies among body parts. More specifically, in the case of a potter for the creation of a clay vase, the right and the left hand have to work synergistically creating inter-limb synergies, in order for the vase to be completed, while in the case of a glass-blower, apart from synergy, the different parts of his limbs support each other, creating a relationship that could be defined as intra-limb mediation. In these two cases, as well as in those of a silk-weaver or a worker in industry, all the body part relationships mentioned above do exist, but it is also obvious that there is another relationship that could be defined as intra-joint association. The joint characteristics (Cartesian coordinates, angles etc.) affect each other inevitably, and because of the inertia effect, every characteristic also depends on its own history (transitioning). All those dependencies and relationships can be defined as four assumptions that can be used to describe human movement through the equation of the Gesture Operational Model (GOM) below.

$$GOM: (intra - joint\ association) + (transitioning) + (inter - limb\ synergies) + (intra - limb\ mediation) \quad (1)$$

The purpose is through the GOM, the creation of a full body model that will be able to represent and describe the movement of the human body. It is assumed that each of the assumptions of ‘intra-joint association’, ‘transitioning’, ‘intra-limb synergies’ and ‘intra-limb mediation’, contribute at a certain level to the production of the gesture. As far as the intra-limb mediation is concerned, it can be decomposed into the ‘inter-joint serial mediation’ and the ‘inter-joint non-serial mediation’.

The proposed model works perfectly for all three dimensions (X, Y, and Z), but for simplicity reasons, it will be presented only for two dimensions, the X and Y. In addition to this, only positions are used, but the model is designed to be able to receive joint angles as input as well. The GOM consists of the four assumptions as defined below.

Assumption 1: Intra-joint assumption

It is hypothesized that each body joint (i.e. right hand) is represented by the Cartesian coordinate system. This means that each motion is decomposed in X and Y coordinates for the spatial part, thus described by two mutually depended variables. It is assumed that there is a bidirectional relationship between the two variables defined as intra-joint assumption.

Assumption 2: Transitioning

It is also assumed that each variable depends on its own history, also called inertia effect. This means that the current value of each variable depends on the values of previous times, also called lag or dynamic effect, which is defined here as transitioning.

Assumption 3: Inter-limb synergies

It is assumed that some entities work together in order to create a final object. We can take as an example the case of a potter trying to create a clay vase. The two hands are not independent but work synergistically to produce the final object and give a symmetrical shape to it (inter-limb synergies).

Assumption 4: Intra-limb mediation

The assumption of mediation can be separated into two sub-assumptions, concerning the serial and non-serial parts of it, the inter-joint serial mediation and the inter-joint non-serial mediation.

Assumption 4.1: Inter-joint serial mediation

It is assumed that a body entity may depend on its neighbouring entities to which it is directly connected to, e.g. a glassblower, while using the pipe, moves his/her wrists along with his/her shoulders and elbows. In case this assumption is statistically significant there is an inter-joint serial mediation.

Assumption 4.2: Inter-joint non-serial mediation

It is assumed that each body entity depends on non-neighbouring entities of the same limb, e.g. the movement of the wrist may depend on the movement of the elbow and shoulder. Thus, it is highly likely that both direct and indirect dependencies simultaneously occur in the same gesture. $Entities$

are named after the first letters of the respective body joint. More specifically, LSH and RSH represent the left and right shoulder respectively. Accordingly, LELBOW and RELBOW represent the left and right elbow, LWRIST and RWRIST, the left and right wrist, LHAND and RHAND the left and right hand. HEAD, NECK and HIPS represent, as their names indicate, the head, the neck and the hips.

So, an example of the representation of those assumptions for the X-axis would be as below:

$$Entity_{1,X}(t) = Entity_{1,Y}(t-1) + Entity_{1,X}(t-1) + Entity_{1,X}(t-2) + Entity_{2,X}(t-1) \quad (2)$$

The assumptions presented above, can lead to a full body movement modelling, as shown in Figure 3. For a better understanding of the figure, the body joints names consist of the first letter of the respective body joint.

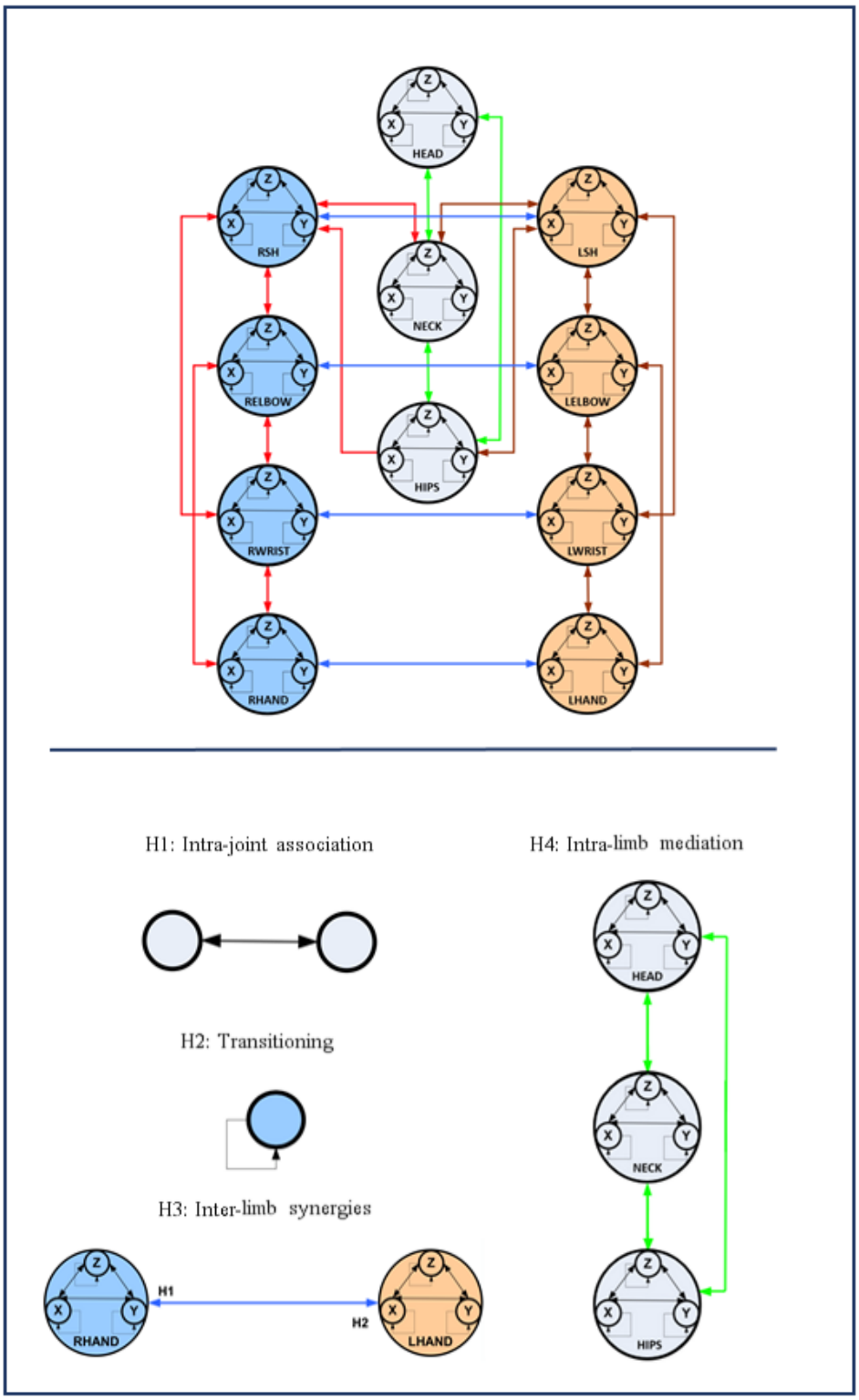

Figure 3. Full body assumptions [37]

3.3.2. Simultaneous equation system

The movement modelling presented above was the basis in the State Space model used. The State Space method permits the depiction and representation of a system, in this case of human movement, for trajectories forecasting.

The State Space representation is created according to the assumptions presented above and is as follows. For simplicity and better understanding, the representation is in one of the three dimensions used. Respectively for the rest two dimensions. The definition of the equations of the system follows the theory of the SS modelling, which gives the possibility for the coefficients to dynamically change over time. A SS model for $n-$dimensional time series $y(t)$, consists of a measurement or observation equation relating the observed data to an $m-$dimensional state vector $s(t)$ and a Markovian state or transition equation that describes the evolution of the state vector over time. The state equation depicts the dependence between the system's past and future and must 'canalize' through the state vector. The measurement or observation equation is the 'lens' (signal) through which the hidden state is observed and it shows the relationship between the system's state, input and output variables. Representing a dynamic system in a SS form, allows the state variables to be incorporated into and estimated along with the observable model.
Therefore, given an input $u(t)$ and a state $s_S(t)$, a SS gives the hidden states that result to an observable output (signal). A general SS representation is as follows:

$$\frac{ds_s}{dt} = As_S(t-1) + w(t) \quad (3)$$

$$y = C\frac{ds_s}{dt} + Du \quad (4)$$

where (3) is the state equation, which is a first-order Markov process (4) is the measurement equation, s_S is the vector of all the state variables, $\frac{ds_S}{dt}$ is the time derivative of the state vector, u is the input vector, y is the output vector, A is the transition matrix that defines the weight of the precedent space, C is the output matrix and D is the feed-through matrix that describes the direct coupling between u and y, and t indicates time.

When capturing the gestures with motion sensors, Gaussian disturbances are also added in both the state and the output equation. After performing the experiments presented in this work, it was observed that Gaussian disturbances didn't change at all the final estimation result, so they were considered to be negligible.

The SS representation of the positions on the X-axis for a body $Entity_{i,j}$ -where i represents the body part modeled in a SS form and j the dimension of each $Entity$- according to the GOM is structured as follows:

$$\frac{ds_s}{dt} = A * s_S(t-1) = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \begin{bmatrix} Entity_{1,X}(t-1) \\ -Entity_{1,X}(t-2) \end{bmatrix} = \begin{bmatrix} \alpha_1 Entity_{1,X}(t-1) \\ -\alpha_2 Entity_{1,X}(t-2) \end{bmatrix} \quad (5)$$

$$\overset{(5)}{\Rightarrow} Entity_{1,\mathrm{X}}(t) = [1\ 0]\frac{ds_s}{dt} + \alpha_3 Entity_{1,\mathrm{Y}}(t-1) + \alpha_4 Entity_{2,\mathrm{X}}(t-1) = \tag{6}$$

$$= \alpha_1 Entity_{1,\mathrm{X}}(t-1) - \alpha_2 Entity_{1,\mathrm{X}}(t-2) + \alpha_3 Entity_{1,\mathrm{Y}}(t-1) + \alpha_4 Entity_{2,\mathrm{X}}(t-1)$$

Where α_i, the coefficients that need to be estimated. In equation 6, $Entity_{\mathrm{X}}(t-2)$ is subtracted by $Entity_{\mathrm{X}}(t-1)$, indicating difference between successive levels of dimensions, e.g. positions on Y-axis (transitioning assumption). Equations 5 and 6 occur by equations 3 and 4 respectively. More specifically, equation 6 consists of the exogenous variables to which the endogenous ones, coming up from the state equation (equation 5), are added. As an example, the SS representation for the right wrist is given:

$$\mathrm{RWRIST_X(t)} = \alpha_1 \mathrm{RWRIST_X(t-1)} - \alpha_2 \mathrm{RWRIST_X(t-2)} + \alpha_3 \mathrm{RWRIST_Y(t-1)} + \alpha_4 \mathrm{LWRIST_X(t-1)} \tag{7}$$

In equation 7, $\mathrm{RWRIST_X(t-1)}$ and $\mathrm{RWRIST_X(t-2)}$ are the endogenous variables, while $\mathrm{RWRIST_Y(t-1)}$, and $\mathrm{LWRIST_X(t-1)}$ are the exogenous ones. The coefficients of the State-Space equations are computed with the use of Kalman filtering, via the method of maximum likelihood estimation.

Kalman filter is an optimal estimator in the sense that if all noise is Gaussian, then the Kalman filter is what minimizes the mean square error of the estimated parameters [32] . It is a recursive method so that new measurements are processed as they arrive. The process of Kalman filtering consists of two recursive steps, the prediction and the update.

Maximum likelihood estimation (MLE) is a method that determines values for the parameters of a model [32] . The parameter values are found so that they maximize the likelihood that a process is described by the model that produced the actually observed data. What we need to ascertain is the total probability of observing all data, for example the joint probability distribution of all observed data points. For this to be done, some condition probabilities need to be calculated, a process that can be a bit complex. The assumption is that all data generated are independent. Like that, the total probability of observing all data comes from observing all data individually.

3.4. Gesture recognition

In this subsection, HMMs will be used for gesture recognition and will accompany the State Space representation as analysed earlier. The dataset used as an input for the recognition engine consists of the gesture below.

Hidden Markov Models (HMMs) have been widely used in recent years for time series recognition, such as voice and gesture recognition. HMMs [25] are a double stochastic process governed by a finite number of states (Markov), where each of the states is associated with a probability distribution. Transitions between states are also governed by a set of probabilities, called transition probabilities. For example, at a discrete time and for a given situation, an observation is generated based on the corresponding probability distribution. Only the observation is evident in the system, without knowing the state of its origin. So, the situation remains "secret", hence the name HMM.

A gesture is a dynamic movement consisting of a sequence of postures. To recognize or rank a sequence of postures data, one can evaluate the probability of emission of this sequence by a set of previously trained HMMs. Data must be available with their respective classes (label). The approach comes down to the following way:

1. Train an HMM with the gesture classes.

2. Evaluate the probabilities of emission of the sequence to be classified for each HMM.

3. Determine, which HMM would most likely have generated the specific sequence of observations, and consequently the class of the gesture.

3.5. Gesture comparison and sonification

Using the hybrid HMMs proposed algorithm that is created by the use of State-Space modelling, the probabilities of the HMMs concerning recognition will be reinforced. When the HMMs do not give the expected good results mostly because of the quality of the dataset, the State Space representation will work as an extra validation step for gesture recognition.

For the evaluation of the performance and the proposed methodology, the metrics $precision$, $recall$ and $f-score$ were calculated. Those metrics are defined as shown below.

$$precision = \frac{\#(true\ positives)}{\#(true\ positives) + \#(false\ positives)} \tag{21}$$

$$recall = \frac{\#(true\ positives)}{\#(true\ positives) + \#(false\ negatives)} \tag{22}$$

$Precision$, $recall$ and $f-score$ are calculated for all the gestures that each gestural vocabulary consists of. For a gesture of class i, $\#(true\ positives)$ represent the number of gestures of class i that were recognized correctly, $\#(false\ positives)$ represent the number of gestures that didn't belong in class i and they were recognized from the algorithm as parts of class i. Finally,

$\#(false\ negatives)$ represents the number of gestures belonging to class i that were not recognized as part of it.

More precisely, $precision$ represents the rate of gestures that really belong in class i, among those who are recognized as class i, while $recall$ represents the rate of iterations of gestures of class i that have been recognized as class i. A measure that combines both precision and recall is the $f-score$, which is given by equation (23).

$$f-score = 2\frac{precision * recall}{precision + recall} \tag{23}$$

The dataset consists of 4 different gestures with 35, 34, 21 and 27 repetitions respectively. 5-20 hidden states were used for training the gesture recognition algorithm, the number of which were again computed for every iteration in the resampling phase. The joints selected for training were the wrist, elbow and shoulder joints for each hand, along with the neck. $Precision$ appears improved in almost every observation and maximum likelihood. The $recall$ in almost every gesture has remained stable except from the third one, where it was increased by +4%.

The mean $f-scores$ and total accuracy for the used dataset is presented. The total accuracy for the dataset has reached 80.34% from 70.94%.

Table 2: Gesture recognition confusion matrix using HMMs and hybrid HMMs approach [37] .

	HMM_1	HMM_2	HMM_3	HMM_4	Recall (%)
G_1	31	2	1	1	88.57
G_2	0	33	1	0	97.05
G_3	2	2	16	1	76.19
G_4	0	0	0	27	100
Precision (%)	93.93	89.18	88.88	93.1	
	$\mathrm{HMM}_1^{\mathrm{SS}}$	$\mathrm{HMM}_2^{\mathrm{SS}}$	$\mathrm{HMM}_3^{\mathrm{SS}}$	$\mathrm{HMM}_4^{\mathrm{SS}}$	Recall (%)
G_1	31	2	1	1	88.57
G_2	0	33	1	0	97.05
G_3	1	1	17	2	80.95
G_4	0	0	0	27	100
Precision (%)	96.87	91.66	89.47	90	

Table 3: Mean f-score and total accuracy using comparing the HMMs approach to the hybrid HMMs proposed [37] .

Mean f-score	HMM	90.64 %
	HMM^{SS}	91.57 %
Total accuracy	HMM	91.45%
	HMM^{SS}	92.3%

The gestures of the glassblower, as they were performed by the user of the cultural installation, are being recognized and then, sonified with the method of explicit mapping. Each gesture is mapped to two different music samples, each one corresponding to the movement of each hand. The gestures to be recognized are 4 in total, mapped to 2 music samples each, thus 8 sound samples in total. The user is asked to perform the 4 glassblowing gestures one by one, in order to reach to the final stage, that of the creation of a glass carafe. A camera captures the performed gesture and sends as input the coordinates of the wrists of the right and left hand to the proposed hybrid HMMs algorithm.

At this point of the project, the quality of the sonification in terms of how close the pitch (for the Y coordinate of each wrist) and panning (for the X coordinate of each wrist) is to that of the original sounds, has to do with how close is the performed gesture to the one of the experts. The sounds mapped to each gesture, are layered creating a complete music piece at the end of the gestural performance. So, the sonification gives a motivation to the user to complete all the tasks/gestures, by also reaching to a musical goal. Following the tempo of the sounds is also a helpful feedback on how well the gestures have been performed and in which way they need to be improved. The process starting from motion capturing, up to the sonification of the gesture is presented in Figure 4.

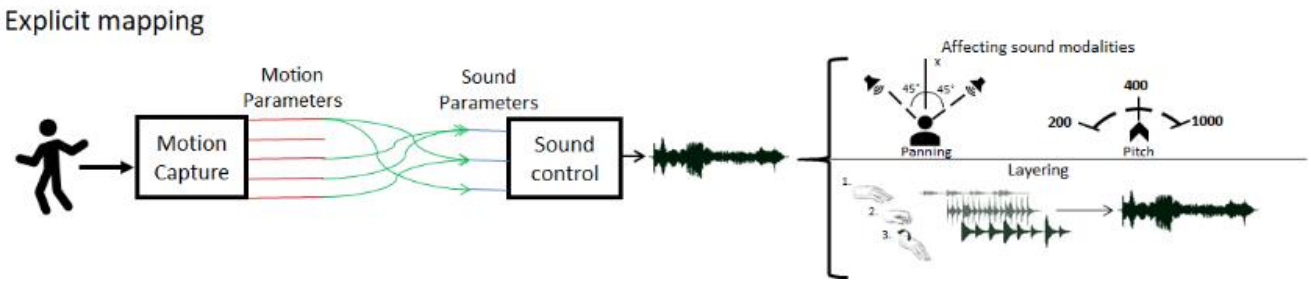

Figure 4: Explicit mapping [38]

The next goal is to extend the existing installation in order to use both implicit and explicit mapping. At this second stage, the recognition results will be also exploited, reinforcing the cultural experience of the user.

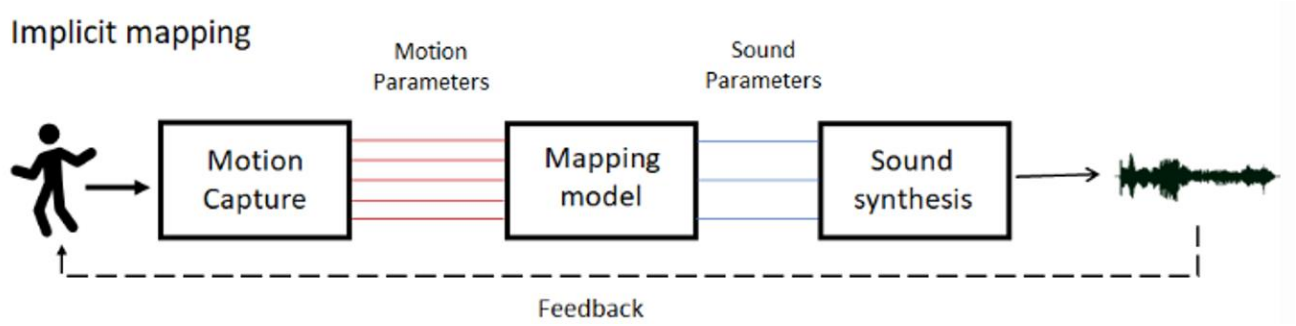

Figure 5. Implicit mapping [39]

For a better understanding of the proposed methodology for sonification, a video was created[3]. A set up simulating the glassblower's equipment has been created in the lab and it is shown in the figure 6. The installation user is asked to perform one by one the gestures that the routine of the glassblower consists of. First, the user is invited to observe the expert gestures, as they are presented in the video and to hear the sonification result of his movements. This is what we call here "the original sounds". A mapping has been done between motion parameters and acoustic features: The tempo modality is affected by the movement of the right and left hand in the x axis, while the panning of the sound is affected by the movement on the y axis. The gestures of the installation user are being recognized and also sonified. The quality of the sonification in terms of how close the tempo or the pitch is to the original sounds has to do with how well the gestures have been performed and recognized. The user is able to perform the gestures one by one, until he reaches to the final gesture, thus the creation of the glass carafe. The sounds mapped to each one of the gestures, are layered creating a complete music piece at the end of the gestural performance. So, the sonification gives a motivation to the user to complete all his tasks/gestures by also reaching to a musical goal. Following the tempo of the sounds is also a helpful feedback on how well the gestures have been performed and in which way they need to be improved. Below in Figures 6, 7 and 8, the stages of the user performance are being shown.

[3] https://www.youtube.com/watch?v=YHf_aMCRVFY&feature=youtu.be

Figure 6: Gesture 1, the user is trying to perform the exact gesture of the expert glassblower.

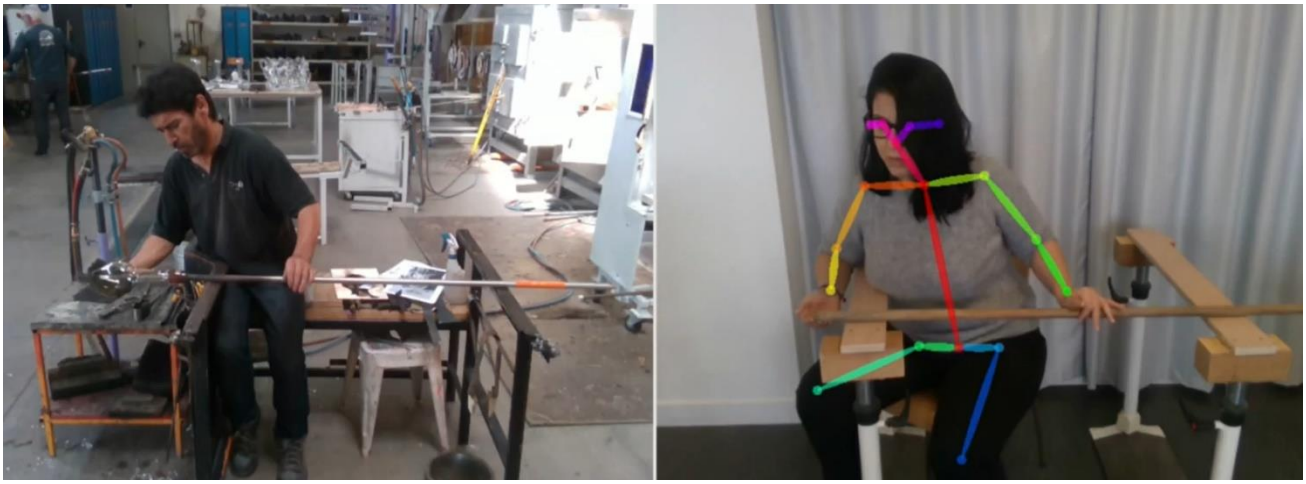

Figure 7: Gesture 2 of the glassblowing routine.

Figure 8: Gesture 3 of the glassblowing routine.

4. Multimodal interaction techniques for Mixed Reality experiences

This section presents a natural interaction infrastructure, called Nibbler, which supports a number of alternative natural interaction techniques. Nibbler builds on the **Microsoft Kinect sensor** and was developed using the **C# programming language, Microsoft Kinect software development kit (SDK) v1.8 and the .NET Framework**. The GUI of the proposed software platform is presented in Figure 10. Nibbler exposes the entire functionality offered by the **Microsoft Kinect software development kit (SDK) v1.8.** In the context of Mingei, currently, the Gestures and postures recognition part of the infrastructure has been fine-tuned and integrated to the Mingei workflow for facilitating user based interaction with the UI of the Mixed Reality surface.

Other modalities, such as Speech Recognition are also available but not foreseen to be used at least due to the current version of use cases, UI designs and concepts definitions. In the case of Speech Recognition this happen through the Microsoft Speech API and thus a plethora of languages is supported, if needed.

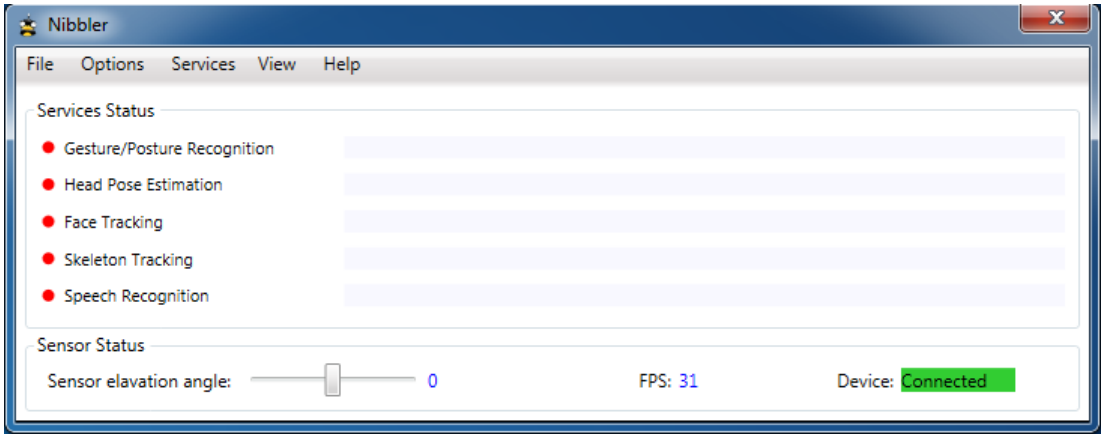

Figure 10. Nibbler UI [40]

4.1. Rationale

The Mingei Mixed Reality use cases as presented in the current state of the project will involve a user interacting with tools in front of a virtual surface. The virtual surface itself does not contain any specialised equipment and thus interaction should happen through tool manipulation and through user gestures in a natural interaction paradigm.

This poses several challenges to the recognition part of the Gesture and Action recognition infrastructure of Mingei for the following reasons:

- Interaction and tracking of interaction with physical objects is a challenging research topic

- Combination of object and gesture based interaction makes the required technology even more challenging

To this end Mingei has taken the technical decision to distinguish object based interaction and natural interaction by providing two trackers running simultaneously in the same equipment and switching between them on the fly based on the UI requirements. This will allow Mingei to exploit mature technologies and research outcomes for the natural interaction part while investing more effort to object based interaction.

In this context Nibbler infrastructure is intended to support the natural part of the interaction with the UI elements of the Mixed Reality (MR) surface implemented in D6.3 and will run in conjunction to the Gesture/Posture recognition techniques presented in this deliverable. The distinction between these modalities will allow the system to depend on different sub-system for each task thus switching between modalities to reduce error rates, improve effectiveness and enhance user experience.

4.2. Sensory modules

Nibbler is organized into various modules, each of which is responsible for specific sensory requirements. Each module is presented in detail in the following sections.

4.2.1. Skeleton tracking module

The skeleton tracking module is responsible for reporting position information of each skeleton joint. This module performs geometric transformations on each skeleton joint position constituting every real time skeleton frame. This happens in order to get the same valid results regardless of the position of the user who may be located everywhere inside the sensor's field of view.

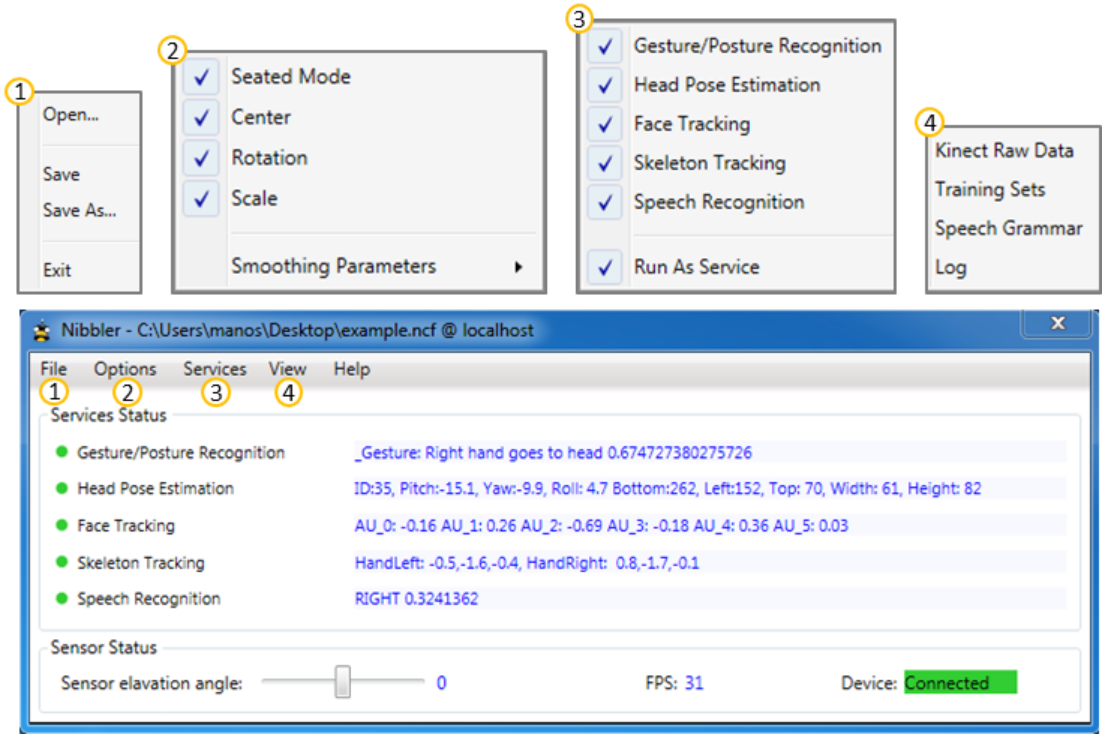

Figure 11. Nibbler in action and UI decomposition [41] .

The skeleton-tracking module transforms the user's skeleton to a local scope, i.e., expressed relatively to the 3-axis coordination system cantered in the middle of the user's shoulders. The transformation is applied with respect to three distinct steps, translation, scaling and rotation, as shown in Figure 11. Firstly, each joint's x-axis position is subtracted with the position dynamically calculated as the centre of both shoulders. This way, skeleton tracking is performed regardless of the user's relative position to the sensor, as presented in Figure 12. Secondly, the joints' positions are normalized in order to be scale-independent. Finally, the module rotates the skeleton so as to align the user's skeleton to the sensor. This is accomplished by multiplying each joint's position with a matrix calculated from the angle θ, where θ is equal to –yaw (yaw is the angle of line between the right and the left shoulder).

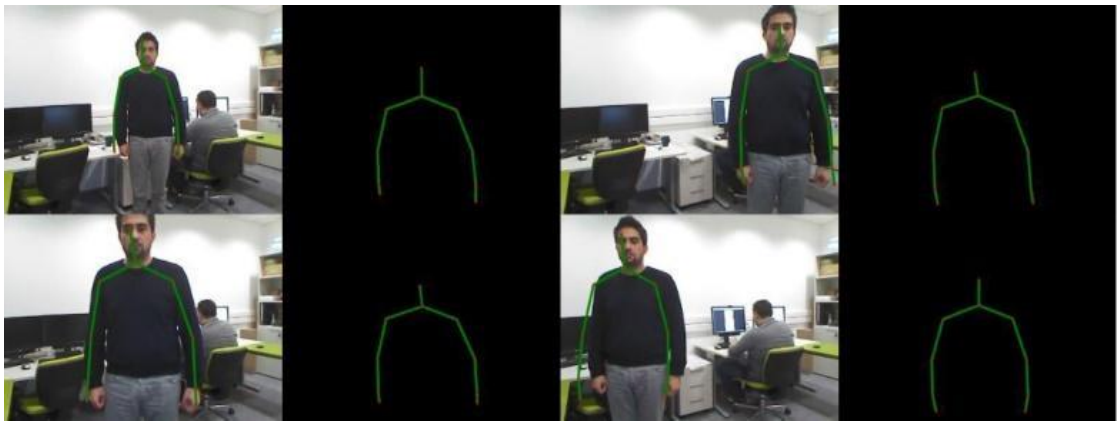

Figure 12. An example of position independence between user and sensor [42] .

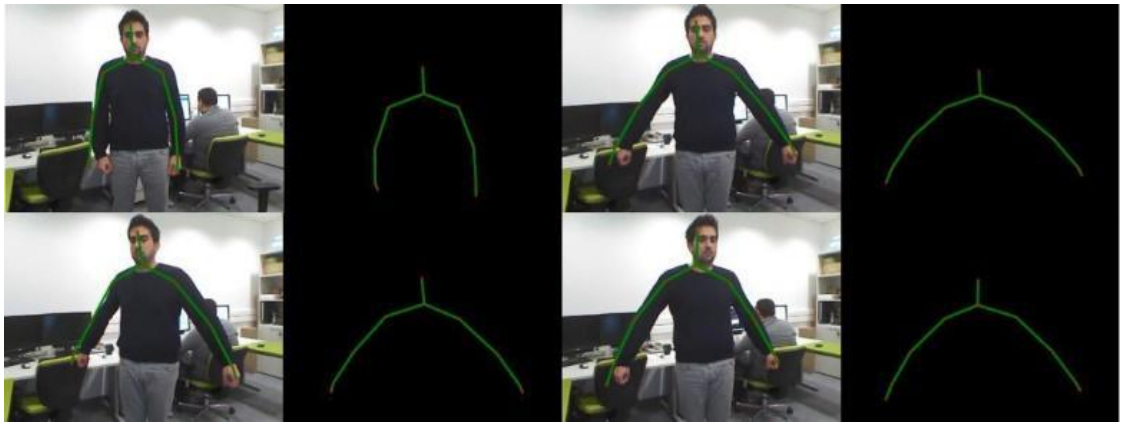

Figure 13. An example of alignment independence between user and sensor [43] .

4.2.2. Gesture/Posture recognition module

Although much work has been done in the domain of Gesture/Posture recognition, there are no ready to use solutions widely available to developers aiming to incorporate Kinect based gesture recognition in their applications. However, the Microsoft Kinect SDK provides a concrete example containing sample source code for the next and previous gestures. To this end, this part of the module partially replicates existing works in gestures recognition building on well-established practices in the domain so as to integrate this very important form of natural interaction to the developed framework. The Gesture/Posture recognition module implements the dynamic time warping (DTW) algorithm for measuring similarity between two skeleton sequences which may vary in time or speed. The most important contributions of this module is the provision of a training platform that allows developers to fine tune their gestures by having access to a number of alternative biometric parameters. In general, the DTW algorithm can be applied to any data which can be turned into a linear sequence. A well-known application has been automatic speech recognition, to cope with different speaking speeds[4]. The first skeleton sequence is captured and fine-tuned only once during the training process, while the second one is captured constantly in real time. During the training process, the author of a gesture is able to record a skeleton sequence using his own body as input data and store it in the database (see Figure 14). The number of the frames in a sequence may vary from 1 up to a maximum predefined variable, which is usually 30 considering that 30 is the maximum sensor's frame rate according to its specification details.

[4] http://en.wikipedia.org/wiki/Dynamic_time_warping

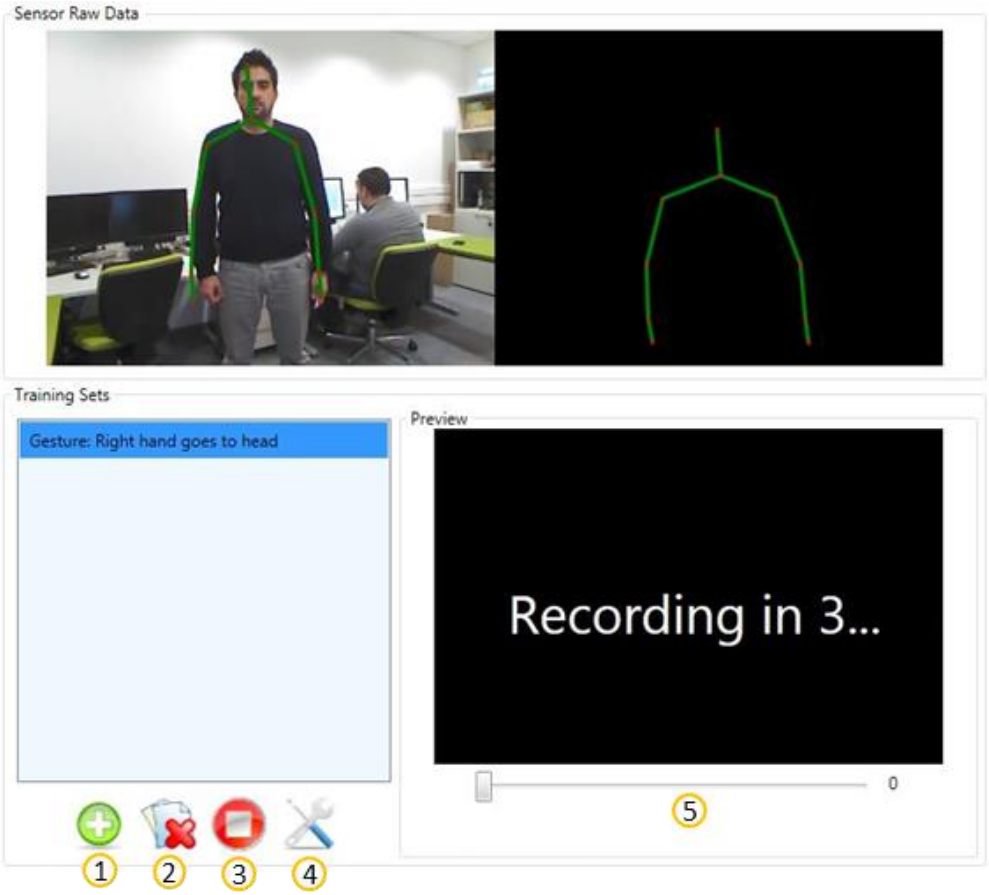

Figure 14. The author starts recording a skeleton sequence in seated mode (i.e. only the upper half skeleton captured) [44]

End users are provided with a plethora of functions to optimally adjust gesture/posture recognition. In particular, end users may (see Figure 14) add a new gesture using button (#1), delete an existing one using button (#2), or start/stop skeleton sequence recording using (#3). When recording is completed, the author is able to preview the recorded skeleton sequence, edit it and fine tune the captured skeleton sequence by pressing button (#4) and using the pop up configuration window as shown in Figure 14. This window offers functionality for: a) renaming (#1), b) adjusting the maximum distance with the real time skeleton sequence (#2) for successful recognition (see below for further details), c) modifying the number of the minimum frames (#3) that have to be captured in real time before the recognition process is triggered, d) trimming the corresponding skeleton sequence (#5), (#6) and e) adjusting some of the basic parameters used in the DTW algorithm such as the slop constraint which determines the maximum slope in the optimal path (#7). Additionally, the author can select only the joints which mainly characterize a gesture (#8), i.e., when the goal is to recognize a gesture in which the user uses his right hand to select the next photo by slightly moving it from right to left, the remaining joints of the body do not need to be taken into account. Furthermore, if some axis doesn't play an important role for a gesture, such as the Z axis (the axis of depth) in the aforementioned example, the author can disable it by

unselecting the corresponding checkbox (#4). Lastly, a gesture playback panel (#9) is available to allow the author preview the recorded skeleton sequence in front and side realization.

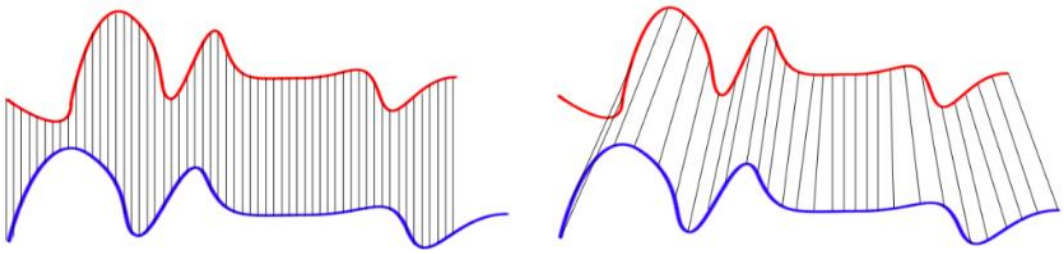

Figure 15: Euclidean vs. Dynamic Time Warping Matching [45]

When the recognition module is running, it captures constantly, in real time, skeleton frames, and when their total number reaches the number equivalent to one second, it then starts the matching process. The latter calculates an optimal distance between the real time sequence and every sequence which is stored in the database. The sequences are "warped" non-linearly in the time dimension to determine a measure of their similarity independent of certain non-linear variations in the time dimension. Dynamic Time Warping (DTW) allows elastic shifting in the time domain and matches sequences that are similar but out of phase as shown in Figure 15.

4.2.3. Using Nibbler from a developer's perspective

The presented modules integrated in the Nibbler sensory infrastructure run simultaneously without any performance issues at almost 30fps on an ordinary pc (see Figure 11). Figure 11 illustrates the settings that are available for configuring Nibbler for the desired context of use. Nibbler communicates with clients and reports each module's measurements via a tcp-ip socket. Additionally, Nibbler can accept requests from clients in real time to change either the gesture training set or the grammar used for speech recognition. In this context, Nibbler's functionality is described in the interface definition language (IDL[5]) as follows:

Definitions	`enum JointTrackingState { Inferred, NotTracked, Tracked};`
	`enum JointType {`
	`HipCenter,`
	`Spine,`
	`ShoulderCenter,`
	`Head,`
	`ShoulderLeft,`
	`ElbowLeft,`
	`WristLeft,`
	`HandLeft,`
	`ShoulderRight,`
	`ElbowRight,`
	`WristRight,`
	`HandRight,`

[5] http://en.wikipedia.org/wiki/Interface_description_language

	HipLeft,
	KneeLeft,
	AnkleLeft,
	FootLeft,
	HipRight,
	KneeRight,
	AnkleRight,
	FootRight
	};
	struct Point3D {
	double X;
	double Y;
	double Z;
	};
	struct Joint {
	JointTrackingState trackingState;
	JointType type;
	Point3D position;
	};
	struct SemanticResultValue {
	string phrase;
	string value;
	};
	typedef sequence<SemanticResultValue> SemanticResultValueSeq;

Gesture recognition	ami::StringSeq GetGestureNames (); boolean LoadGestures(in ami::OctetSeq gesturesConfigStream); void Event_GestureRecognized (in string gesture, in double distance); void Event_GestureRecognizedExt (in string gesture, in double distance, in ami::OctetSeq colorImgStream);
Head pose estimation	void Event_HeadPoseChanged (in double pitch, in double yaw, in double roll); void Event_HeadRectChanged (in long left, in long bottom, in long top, in long width, in long height);
Face tracking	void Event_FaceAnimationUnitsChanged (in ami::FloatSeq faceAnimationUnits);
Skeleton tracking	void Event_HandRightPositionChanged (in Point3D position); void Event_HandLeftPositionChanged (in Point3D position); void Event_SkeletonChanged (in JointSeq joints); void Event_SkeletonChanged (in JointSeq joints);

4.3. Evaluation through the Mimesis game

The Mimesis game is an existing software product of FORTH and was selected as a heuristic evaluation platform of the toolkit. The evaluation targeted the validation of the implemented technology in terms of interaction in order to ensure that sufficient results are received through the recognition infrastructure. This validation is not related with the user based evaluation of the final

Mingei experiences that will happen in the context of the pilots' realisation. On the contrary it regards the validation of technology that will be used for the formulation of the pilots.

The Mimesis game requests the user to assume various body postures as illustrated in Figure 17 and due to this fact it is a very good testbed for measuring recognition rates. Using the Nibbler sensory infrastructure, the game measures the quality and performance of the body posture. In detail, the user stands in front of a display, which is positioned horizontally, (e.g., near to a wall) giving the feeling of standing in front of a mirror. At the upper side of the large display, a depth camera (i.e., Kinect sensor) is positioned, which allows Nibbler to recognize the user position and his gestures as well. The sensor sets restrictions regarding the view area that ranges from 1.2 m to 4 m and in around 45 degrees viewing angle. The Mimesis game requires the user to imitate a series of postures demonstrated by the system (see Figure 17).

Figure 16. Mimesis game gestures [46]

The Mimesis game randomly selects a different posture. When all postures have appeared the game ends. Figure 18 presents some screenshots where the user is assuming the pose indicated by the VC.

Figure 17. A short compilation of screenshots during playing Mimesis game [47]

4.4. Preliminary evaluation results

In general, the evaluation of the Nibbler sensory infrastructure in the context of the Mimesis game proved that users (developers, part of the team at FORTH) where generally satisfied by the recognition rate of the system. The main shortcoming that rose was that in some cases users were unable to keep an optimal distance from the large display. This had as a result the inability of Nibbler's skeleton tracking module to recognize their appearance due to the limitations set by the sensor device (i.e., Kinect sensor has a practical ranging limit of 1.2–3.5m).

Thus it is recommended for Mingei installations that visual cues to the virtual world and the physical world exist so as to provide an indication regarding whether the user is assuming a correct position within the viewing frame of the sensor. Thus the users will have visual feedback regarding their position (e.g. green footprints that appear in the virtual world) and when they move too close or too far these may turn to orange and then red. Finally, these visual cues may disappear when the user is out of range.

4.5. Discussion

In this section the sensory infrastructure mechanism, called Nibbler was presented together with some initial experiment to evaluate the functionality of the game in the context of real usage scenarios.

To this end an existing Virtual Space was used where the user should position him-self and assume several postures.

With this scenario Mingei simulated the usage of the infrastructure in the context of the Mingei Mixed Reality surface where apart of the object based interaction users will be requested to assume certain postures to interact with the user interface of the surface. For example within a workshop (e.g. glassmaking) users will have the possibility to interact with tools to emulate the glassmaker gestures but also assume postures to interact with the UI of the workshop (e.g. Hands up: pause experience, hands down: resume, etc.).

The effectiveness of user interaction and of the selection of gestures and postures will undergo a user based usability and user experience evaluation during the pilot realisation.

5. First experimental results of movement sonification and future works

First preliminary results of movement sonification have been achieved based on the methodology described in previous sections. A gestural vocabulary has been defined by each cultural partner, as described in D5.1 and the data are processed in order to be used for the definition of the gesture operational model and to be then compared with the gesture of the final user. A first version of bounds (minimum and maximum) per gesture performed by the expert has been defined. A first version of a module permitting to map a certain number of motion parameters to sound characteristics have been developed and a first test in lab conditions has been released where a) experts gestures have been simulated, performed and recorded by researchers, b)the same gestures have been reproduced and their parameters have been mapped to sound (movement Sonification). However, no quantitative analysis has been performed yet.

The future work will be focused on the implementation of the methods developed in real-time in order to achieve the comparison between the time-series and the measurement of their deviation. The features and the motion parameters that will be compared should be identified. In parallel the structure of the interaction mechanism (thresholds, rules etc.) that will trigger the sonification will be defined. It is necessary to take into consideration A deeper study of various sonification techniques (mostly explicit and implicit mapping) will be done in order to identify and implement the most appropriate strategy.

References

[1] Sonia Duprey, Alexandre Naaim, Florent Moissenet, Mickaël Begon, Laurence Cheze. Kinematic models of the upper limb joints for multibody kinematics optimisation: An overview. Journal of Biomechanics, Elsevier, 2017, 62, pp. 87-94. ff10.1016/j.jbiomech.2016.12.005ff. ffhal-01635103.

[2] Zalmai Nour, Kaeslin Christian, Bruderer Lukas, Neff Sarah, Loeliger Hans-Andrea, "GESTURE RECOGNITION FROM MAGNETIC FIELD MEASUREMENTS USING A BANK OF LINEAR STATE SPACE MODELS AND LOCAL LIKELIHOOD FILTERING", IEEE 40th International Conference on Acoustics, Speech and Signal Processing, ICASSP, Brisbane, Australia, April 19-24, 2015

[3] Michał Lech and Bożena Kostek, "Hand gesture recognition supported by fuzzy rules and Kalman filters", Int. J. Intelligent Information and Database Systems, Vol. 6, No. 5, 2012 407

[4] Pedersoli, Benini, Adami, Leonardi, "XKin: an open source framework for hand pose and gesture recognition using Kinect", © Springer-Verlag Berlin Heidelberg 2014, Vis Comput DOI 10.1007/s00371-014-0921-x.

[5] J.K. Aggarwal, Q. Cai, "Human Motion Analysis: A Review" ,Computer Vision and Image Understanding, Volume 73, Issue 3,1999, Pages 428-440,ISSN 1077-3142, https://doi.org/10.1006/cviu.1998.0744.

[6] Vasileios Sideridis, Andrew Zacharakis, George Tzagkarakis and Maria Papadopouli, GestureKeeper: Gesture Recognition for Controlling Devices in IoT Environments, arXiv:1903.06643v1 [cs.HC] 15 Mar 2019.

[7] Yang Ruiduo & Sarkar Sudeep, Gesture Recognition using Hidden Markov Models from Fragmented Observations. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, 1. 766- 773. 10.1109/CVPR.2006.126.

[8] Li, Koping, Schmitz, Grzegorzek, Real-Time Gesture Recognition using a Particle Filtering Approach., ICPRAM 2017.

[9] Jirak, Barros, Wermter, Dynamic Gesture Recognition Using Echo State Networks. ESANN 2015 proceedings, European Symposium on Artificial Neural Networks.

[10] Portillo-Rodríguez,Sandoval-Gonzalez, Ruffaldi, Leonardi, Avizzano, Bergamasco" Real-Time Gesture Recognition, Evaluation and Feed-Forward Correction of a Multimodal Tai-Chi Platform",HAID,2008.

[11] Vaitkevičius, Taroza, Blažauskas, Damaševičius, Maskeliūnas, Woźniak, Recognition of American Sign Language Gestures in a Virtual Reality Using Leap Motion. Applied Sciences, 2019.

[12] Alexandra Psarrou, Shaogang Gong, Michael Walter, "Recognition of human gestures and behaviour based on motion trajectories" Image and vision computing, 2001.

[13] P. Kolesnik, and M.M. Wanderley. Implementation of the Discrete Hidden Markov Model in Max/MSP Environment. In Proc. of the FLAIRS, 2005, 68-73.

[14] F. Bettens, and T. Todoroff. Real-time dtw-based gesture recognition external object for max/msp and puredata. In Proc. of the SMC 2009 Conference, 30, 35, 2009.

[15] J. Françoise. Motion-sound mapping by demonstration. Ph.D. Thesis, Pierre and Marie Curie University, France, 2015.

[16] B. Caramiaux, N. Montecchio, A. Tanaka, and F. Bevilacqua. Adaptive Gesture Recognition with Variation Estimation for Interactive Systems. ACM TiiS, 4, 4, 2015.

[17] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana. Continuous realtime gesture following and recognition. In Proc. of the 8th International Conference on Gesture in Embodied Communication and Human-Computer Interaction, Bielefeld, Germany, 2009.

[18] F. Bevilacqua, R. Muller, and N. Schnell. MnM: a Max/MSP mapping toolbox. In Proc. of the NIME'05, Vancouver, Canada, 2005.

[19] F. Bevilacqua, F. Guédy, N. Schnell, E. Fléty, and N. Leroy. Wireless sensor interface and gesture-follower for music pedagogy. In Proc. of the NIME'07, New York, NY, 2007, 124-129.

[20] A.F., Bobick, and A.D. Wilson. A state-based approach to the representation and recognition of gesture. IEEE TPAMI, 19, 12, 1997, 1325-1337.

[21] J. Françoise. Motion-sound mapping by demonstration. Ph.D. Thesis, Pierre and Marie Curie University, France, 2015.

[22] Christina, Volioti & Manitsaris, Sotiris & Katsouli, Eleni & Manitsaris, Athanasios. "x2Gesture: how machines could learn expressive gesture variations of expert musicians.", 2016.

[23] Baptiste Caramiaux, Nicola Montecchio, Atau Tanaka, Frédéric Bevilacqua. Adaptive Gesture Recognition with Variation Estimation for Interactive Systems. ACM Transaction on Interactive Intelligent Systems, 2014, 4 (4), pp.35. ff10.1145/2643204ff. ffhal-01266046f

[24] Françoise, J. (2013). Gesture–Sound Mapping by Demonstration in Interactive Music Systems. Proceedings of the 21st ACM International Conference on Multimedia (MM'13). Barcelona, Spain, 1051-1054. doi:10.1145/2502081.2502214.

[25] L.R. Rabiner (1989): "A tutorial on Hidden Markov Models and selected applications in speech recognition." *Proceedings of the IEEE* **77**: 257–286.

[26] Christina, Volioti & Manitsaris, Sotiris & Hemery, Edgar & Charisis, Vasileios & Hadjileontiadis, Leontios & Hadjidimitriou, Stelios & Katsouli, Eleni & Moutarde, Fabien & Manitsaris, Athanasios. (2018). A Natural User Interface for Gestural Expression and Emotional Elicitation to Access the Musical Intangible Cultural Heritage. Journal on Computing and Cultural Heritage. 11. 10.1145/3127324.

[27] Françoise, J., "Motion-sound mapping by demonstration", Doctoral dissertation,2015.

[28] Sigrist, R., Rauter, G., Riener, R., & Wolf, P. (2013). Augmented visual, auditory, haptic, and multimodal feedback in motor learning: a review. Psychonomic bulletin & review, 20(1), 21-53.

[29] A. Hunt, M. M. Wanderley, and R. Kirk, "Towards a Model for Instrumental Mapping in Expert Musical Interaction," in Proceedings of the 2000 International Computer Music Conference, 2000, pp. 209–212.

[30] Arfib, D., Couturier, J. M., Kessous, L., & Verfaille, V. (2002). Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces. *Organised Sound, 7*(2), 127-144. doi:http://dx.doi.org/10.1017/S1355771802002054.

[31] Christina, Volioti & Manitsaris, Sotiris & Hemery, Edgar & Charisis, Vasileios & Hadjileontiadis, Leontios & Hadjidimitriou, Stelios & Katsouli, Eleni & Moutarde, Fabien & Manitsaris, Athanasios. (2018). A Natural User Interface for Gestural Expression and Emotional Elicitation to Access the Musical Intangible Cultural Heritage. Journal on Computing and Cultural Heritage. 11. 10.1145/3127324.

[32] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, and Raj Reddy. Spoken Language Processing: A Guide to Theory, Algorithm, and System Development. Prentice Hall PTR, USA, 1st edition, 2001. (15)

[33] Cao, Hidalgo, Simon, Wei, Sheikh. "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields", 2018.

[34] Gavriela Senteri, Sotiris Manitsaris, Alina Glushkova. Methodology pipeline- From gesture recognition to gesture sonification, (2020), Bibliographical reference: Unpublished research work.

[35] Gavriela Senteri, Dimitrios Makrygiannis, Sotiris Manitsaris, Alina Glushkova. The glassblowing gestural vocabulary, (2020), Bibliographical reference: Unpublished research work.

[36] Gavriela Senteri, Dimitrios Makrygiannis, Sotiris Manitsaris, Alina Glushkova. Representation of the body dependencies observing the gestural routine of a glassblower and a silk-weaver, (2020), Bibliographical reference: Unpublished research work.

[37] Sotiris Manitsaris, Gavriela Senteri, Dimitrios Makrygiannis, Alina Glushkova, Human movement representation on multivariate time series for recognition of professional gestures and forecasting their trajectories, 2020, Hypothesis and Theory, Frontiers in Robotis and AI-Sensor Fusion and Machine Perception.

[38] Gavriela Senteri, Dimitrios Makrygiannis, Sotiris Manitsaris, Alina Glushkova. Explicit mapping, sonification methodology, (2020), Bibliographical reference: Unpublished research work.

[39] Gavriela Senteri, Dimitrios Makrygiannis, Sotiris Manitsaris, Alina Glushkova. Implicit mapping, sonification methodology, (2020), Bibliographical reference: Unpublished research work.

[40] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), Nibbler UI, [Screenshot], Taken from Nibbler UI, Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[41] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), Nibbler in action and UI decomposition, [Screenshot], Taken from Nibbler UI, Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.),

Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[42] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), An example of position independence between user and sensor, [Screenshot], Taken from Nibbler UI, Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[43] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), An example of alignment independence between user and sensor, [Screenshot], Taken from Nibbler UI, Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[44] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), The author starts recording a skeleton sequence in seated mode (i.e. only the upper half skeleton captured), [Screenshot], Taken from Nibbler UI, Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[45] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), Euclidean vs. Dynamic Time Warping Matching, [Graphs], Bibliographic reference: Zidianakis, E., Partarakis, N., Antona, M., & Stephanidis, C. (2014). Building a Sensory Infrastructure to Support Interaction and Monitoring in Ambient Intelligence Environments. In N. Streitz & P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions – Volume 21 of the combined Proceedings of the 16th International Conference on Human-Computer Interaction (HCI International 2014), Crete, Greece, 22-27 June, pp. 519-529. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 8530, ISBN: 978-3-319-07787-1).

[46] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), Mimesis game gestures, [Screenshot compilation], Taken form Mimesis UI, Bibliographic reference: Z Zidianakis, E., Antona, M., & Stephanidis, C. (2018). Activity Analysis (ACTA): Empowering Smart Game Design WITH A

General Purpose FSM Description Language. IADIS International Journal on Computer Science and Information Systems, 13 (1), 82-95. ISSN: 1646-3692. [On-line]. Available at: http://www.iadisportal.org/ijcsis/papers/2018130106.pdf

[47] Zidianakis, E., Partarakis, N., Zabulis, X. (2019), A short compilation of screenshots during playing Mimesis game, [Screenshot compilation], Taken form Mimesis UI, Bibliographic reference: Z Zidianakis, E., Antona, M., & Stephanidis, C. (2018). Activity Analysis (ACTA): Empowering Smart Game Design WITH A General Purpose FSM Description Language. **IADIS** International Journal on Computer Science and Information Systems, 13 (1), 82-95. ISSN: 1646-3692. [On-line]. Available at: http://www.iadisportal.org/ijcsis/papers/2018130106.pdf