Deliverable 5.1.n – Offline human motion capture

Release Date	31.05.2020
Version	1.5
Dissemination Level	Confidential

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

Deliverable Number	D5.1
Deliverable Title	Offline human motion capture
Deliverable Type	Report
Dissemination Level	Confidential
Contractual Delivery Date	November 2019 (M12)
Actual Delivery Date	29/11/2019
Work Package	WP5 - DEVELOP & EXECUTE: Build Toolkit and Platform
Authors	Sotiris Manitsaris, Alina Glushkova, Dimitris Menychtas, Brenda Olivas, Xenophon Zabulis, Nikolaos Partarakis, Hatjiantoniou Antonios
Number of pages (incl. cover)	50

Executive summary

One of the main contributions of Mingei is the digital documentation of the motions performed by the experts using motion capture. This deliverable describes the procedure and gestural

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/

know-how recorded and digitised. The joint angles from a silk weaver, glassblower, and two mastic farmers have been captured with an Inertial Measurement Unit (IMU) full body suit from Nansense Inc. while executing their gestures under real conditions. Simulated gestures without the use of real materials were captured as well from one mastic farmer. Each motion file uses the Biovision Hierarchy format, and count with an XML file with its description and recording information.

For further post-processing of the MoCap files, a BHV editor called Animation Studio is being developed. AnimIO allows visualization, editing, and annotation by isolating animation segments and the associated video. AnimIO was created to address important gaps in the state of the art regarding the segmentation of MoCap and video data as summarised as follows: (a) MoCap editing software are to be used by experts, scientist and computer animators, (b) Video editing software are to be used by video editing experts and (c) there is no software that supports synchronised MoCap and Video editing. Addressing such constraints, the AnimIO is an easy and straightforward platform that can combine all the aforementioned technical requirements is a software with a narrow learning curve as non-technical knowledge is required and the UI metaphors used follow the popular desktop video editing software bundled with modern operating systems (e.g. Windows Movie Maker). For the development of AnimIO we opted using Unity3D[1] game engine because it can easily combine those (2D and 3D) spaces together and it has a straightforward API for interaction between them. The resulting application is a stand-alone desktop application that hides from the user the implementation details of the Unity3D[2] game engine. No editing in unity takes place. For video manipulation and editing (by AnimIO) such as splitting a video to smaller parts or merging video parts together, we needed to integrate an external tool. We ended up using FFMPEG[3] command line tool due to its efficiency and ease of use within Unity. Users are not required to run the FFMPEG executable from the command line. AnimIO automates this process by silently creating a FFMPEG processing thread that carries out the requested tasks without the user knowing that an external software is used. No prior knowledge or expertise on FFMPEG is required by end users and its invocation does not affect the overall user experience offered by the software.

The deliverable is structured as follows:

- **Section 1** provides an introduction to the context and purpose of this deliverable

- **Section 2** provides an overview of motion recording technologies

- **Section 3** provides an overview of the relevant background and related work on MoCap

[1] https://unity.com/
[2] https://unity.com/
[3] https://www.ffmpeg.org/

technologies in general but also, regarding MoCap data reuse and MoCap segmentation technologies. Regarding segmentation technologies although allot of work has been done for automatic segmentation these either rely on pre-existing datasets or are methods for on the wild segmentation based on movement features identification, identification of cyclic activities, etc. In the case of Heritage Crafts were no relevant MoCap data exist Mingei proposes a methodology based on AnimIO that allows the creation of multimodal datasets through manual segmentation of MoCap and video sources for domain experts (curators, ethnographers, etc.). In this context Mingei argues that it can allow CH professional to get the most out of MoCap technology without the need to invest on acquiring specialized technology and involving MoCap and Computer Vision experts. To this end Mingei, proposes. The design and implementation of straightforward usable tool that abstracts and hides technical details from end users while offering a straightforward video editing metaphor for synchronous segmentation and annotation.

- **Section 4** provides a summary of Data collections, processing and delivery to other WPs and activities. More specifically the recording protocol formulated in order to ensure that all MoCap sessions produce uniform data recorded under the same conditions and using the same equipment is presented. Then this section moves into describing the MoCap sessions conducted during the first year of the project for all pilot sites. This process resulted to a set of raw MoCap data that were in turn post processed (filtering, correction of erroneous postures and data segmentation) to produce the final collection of gestures for each craft instance. These were delivered to Mingei technical partners using a homogeneous data format followed by additional information in an xml description file and a common naming convention.

- **Section 5** presents the implementation of Animation Studio (AnimIO) for post-processing of MoCap animation files. In the context of Mingei, video, MoCap files resulting from the process discussed in section 4 and processed MoCap files stemming from visual tracking methods (discussed in D5.4) are available. In this sense, it is important in order to generate multimodal datasets, to propose a methodology to segment synchronously these data streams. Such a tool, in the best of our knowledge, does not exists in the literature or as a commercial product. Under the light of this requirement, the Animation Studio (AnimIO) tool was designed and developed both for post-processing MoCap animation files and for synchronous segmentation of MoCap and video files. This tool provides a project based environment were MoCap and video sources can be loaded, previewed and edited. It uses the video timeline metaphor to synchronize video and MoCap streams and the standard video segmentation approach to segment both video and MoCap data. The result is a straightforward and easy to use application undergone a four stages design and implementation process as described in deliverable D4.2.

- **Section 6** provides conclusions and future work

This deliverable is submitted in the context of T5.1. 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

Motion capture, joint angles, Biovision Hierarchy format, real conditions, gestures, and annotation.

Document History

Date	Version	Author/Editor	Affiliation	Comment
17.10.2019	0.1	Brenda Olivas	ARMINES	Initial draft
5.11.2019	0.5	Xenophon Zabulis, Nikolaos Partarakis, Hatjiantoniou Antonios	FORTH	AnimIO implementation
7.11.2019	0.9	Brenda Olivas	ARMINES	Second draft
7.11.2019	1.0	Dimitrios Menychtas, Alina Glushkova, Sotiris Manitsaris	ARMINES	Final draft
26.11.2019	1.1	Brenda Olivas, Dimitrios Menychtas	ARMINES	Corrected Final Draft
29.11.2019	1.2	Margherita Antona	FORTH	Final QA review
10/05/2020	1.3	Dimitrios Menychtas, Alina Glushkova, Sotiris Manitsaris	ARMINES	Updated after comments received by project monitors
15/05/2020	1.4	Nikolaos Partarakis	FORTH	Updated after comments received by project monitors
30/05/2020	1.5	Dimitrios Menychtas, Alina Glushkova, Sotiris Manitsaris	ARMINES	Final version including Mingei internal review comments

Abbreviations

HdS	Haus der Seidenkultur
CERFAV	Centre Européen de Recherches et de Formation aux Arts Verriers
ARMINES	Association pour la recherche et le development des methodes et processus industriels
FORTH	Foundation for Research and Technology - Hellas
IMUs	Inertial Measurement Units
BVH	Biovision Hierarchy format
XML	Extensible Markup Language
FK/IK	Forward Kinematics/Inverse Kinematics
HCs	Heritage Crafts
CrO	Crafts Ontology

Table of Contents

List of Figures

List of Tables

1. Introduction

Mingei's focus is to preserve three historically important European crafts. Silk-weaving from Krefeld, Germany, glass-blowing from Vannes la Châtel, France, and mastic cultivation from Chios, Greece have been chosen to be part of Mingei's efforts. Those crafts were important for the local society and economy of their respective regions. They require specialized equipment and/or material that are rare outside their region of origin. However, their most fundamental aspect is the motions the expert is performing. As such, recording those motions is extremely important for the scope of Mingei. To make this possible, each craft has been broken down to smaller parts and motion capture (MoCap) equipment was used. Additional processing of the recorded motions was done to ensure the motions were recorded without external noises. The resulting data were used to create the animations necessary by other work packages of Mingei.

2. Recording technologies

For the recording of the motions during each task, the inertial measurement units (IMUs) full-body motion capture suit (NANSENSE-BioMed Bundle, Baranger Studios, Los Angeles, CA, USA) was used. One task (glass-blowing) required the measurement of the breathing volume of one of the experts. For this, a piezoelectric (PZT) respiration sensor (Biosignalplux, Lisbon, Portugal) was utilized. All components are portable, therefore it was possible to record the expert under both real and simulated conditions. The whole body movements were recorded including the hands' gestures.

2.1Motion capture

For the capture of the expert's gestures the Nansense BioMed bundle motion capture system was used. This system consists of a full-body suit composed of 52 IMU sensors placed throughout the body and hands. The sensors allow measuring the articulated spine chain, shoulders, limbs, and fingertips in a rate of 90 frames per second. This method was selected over an optical system because of the limitations presented in the recording environments. For the silk pilot, the workshop was indoors with limited space due to the size of the looms, and the rest of the silk preparation equipment. This was not a problem for the inertial system since the expert needs only to wear the suit. On the other hand, confined places are problematic for an optical system due to possible occlusions. In the Mastic Museum at Chios, the recordings were outdoors under the mastic trees. For these conditions the inertial system is more appropriate due to its compactness, and also because optical systems are sensitive to light changes and reflections. Finally, for the glass-making task, the recordings are in a high-temperature environment, a problem for optical systems because of the infrared radiation emitted by heat sources. In essence, for all of Mingei's pilots, the motion capture system needed to be independent of the environment. Therefore, an IMUs suit was chosen.

2.2Breathing sensor

During the Glass-making recordings for the creation of a glass jug, the molten glass needed to be inflated by an assistant while the expert was creating the object. A piezoelectric respiration sensor was placed on the chest of the assistant to capture this last action. The sensor measures displacement variations induced by inhaling or exhaling through a piezoelectric film.

3. Background and related work

3.1Motion Capture

Motion capture (MoCap) is the process of recording the movement of objects or persons medical, entertainment, scientific, or military purposes. Regardless of the intended use, the output of MoCap is the positions in space of the bodies of interest. A wide variety of technologies is used to record the motions of interest. The background will be presented in this segment; however, a special focus will be given in the way those technologies differ with each other from a technical and practical standpoint. The reason being that all technologies give the same output and it might not be apparent why there are so many different methods.

The most popular, and probably the most refined, method is the marker-based MoCap technique. This technology uses an array of cameras that emit and receive infrared light. They are placed in a way that encircles the space to be recorded giving a 360^{o} view of the scene. The bodies to be recorded, either objects or persons, have retroreflective spherical markers attached on their moving parts. A retroreflective marker will reflect a photon at 0^{o} angle. In practice, the infrared photons emitted by the cameras will return straight back to the lens. This allows the software to triangulate the detected photons and reconstruct the position of the markers in a 3D space. The output is a cluster of markers that move in correspondence to the recorded motions. To extract the motions in a more readable manner, a post-processing pipeline is used to create a skeleton out of the marker cluster and calculate the joint angles. This eventually leads to a stick figure animation that performs all the recorded motions. This technology requires a controlled environment free from obstructions. The setup is not particularly portable since the cameras need to be set up in a specific manner and be connected to the recording and processing hardware. It also needs to be noted that marker-based systems suffer when there is a lot of background infrared light (e.g. direct sunlight, multiple heat sources etc.) and therefore they must be in a relatively isolated room. Once those conditions are met (such as in a lab), marker-based systems are very accurate and extremely flexible, allowing the recording of multiple people and moving objects.

Another widely used MoCap technology that can only be used on humans, is the inertia measurement units (IMUs)-based method. This technique uses a combination of gyroscopes, accelerometers, and magnetometers to record the motion of each segment of the human body. Each recording from an IMU is in a global coordinate frame and the differences between them are used to calculate the joint angles of the human body. The skeleton is usually inputted manually, and the segments are predefined. IMUs are generally susceptible to electromagnetic interferences and they tend to drift over time, giving erroneous angles. Also, this method is intrusive because the person usually has to wear the IMUs that are usually woven into a tight full-body suit. This makes it uncomfortable to record for long durations. Another drawback of this technology is its rigid output. While a marker-based system can record any kind of moving objects and the skeleton will be added during post-process, IMU-based systems can only record in their pre-defined skeleton. As a result, inanimate objects, robots, animals, and people with amputation cannot be recorded. However, this technology is portable. As long as the person fits in the suit, the recordings can take place in any environment regardless of obstructions. Though electromagnetic interferences can still be a problem, the fact remains that an IMU-based system can record in most environments with minimum post-processing. Finally, a MoCap technology that is still relatively early in its development is computer vision (CV)-based motion capture. This method uses standard RGB video with or without depth mapping to record motion. Unlike the previous methods, the equipment that it uses isn't particularly specialized as it requires a standard video camera and can include an infrared depth-mapping camera. The bulk of the motion extraction happens using machine learning algorithms to identify the human body and calculate the joint angles. It needs to be stressed that this technology is an active field of research and as such a comprehensive background cannot be given here. The focus here will be on the limitations and how they can impact Mingei. Currently, the method is susceptible to IR interferences, though modern hardware makes only extreme cases like direct sunlight an issue, it needs an unobstructed view of the people being recorded and can potentially fail if other people enter and leave the field of view in the background, it uses a predefined skeleton that is applied on the subjects, and it generally has a high processing cost. However, the appeal of using a camera to perform MoCap is highly applicable when unhindered motions are required, and the duration of each trial needs to be relatively long. It also needs to be mentioned that there are no standard methods for CV MoCap, but rather some popular research platforms, such as OpenPose [41].

To sum up, all MoCap technologies output a skeleton with corresponding joint angles however, there are a few different ways to arrive at that output. Marker-based MoCap systems are very accurate and they don't have any limitations on what they can record, but they require an isolated environment. IMUs-based systems can only record one able-bodied person but they are not limited by their environment and are portable. CV-based MoCap is not an industry standard yet, making it a risky option for MoCap, this will probably change as the technology matures though.

3.2 Motion Capture data reuse

The reuse of human motion capture data (MoCap data) was traditionally important in computer animation, game industry, medical rehabilitation, etc. In MoCap based animation, segmentation of motion is considered one of the fundamental functions. This is an active research fields that captures the attention of researchers for more than a decade. In this context, most of the research work is targeting automatic segmentation of MoCap data with or without prior knowledge. Prior knowledge may be in the form of MoCap datasets or in the form of assumptions on Human motions.

3.3 Automatic classification of human motion

In this context several researchers have proposed different approaches for the classification of human motion. According to [43] human motion has its own regularities, so reasonable prior assumptions on these regularities will lead to better performance. In their research they focus on the learning of intrinsic regularities of MoCap data based on a small set of training data which only contain daily-life motions. By utilizing these learnt motion regularities, they can successfully segment long motion sequences containing motion types that are not even included in the training data [15].

Another segmentation approach proposed in an advanced zero crossing segmentation technique [17]. As zero crossing performs poorly on weak motion, a threshold is defined detecting phases of week motion. In their work they distinguish two states: One for the resting phase and one for phases of movement. Splitting this second phase again, the presented approach leads to a symbolic level allowing later steps to be carried out without the need of considering spatio-temporal dependencies [17]. A code-word sequence matching approach has been proposed by [18]. In this research a family of pose-histogram-based classifiers is developed to examine the spatial distribution of human poses. Then the analysis of the performance of the temporal and spatial classifiers is done separately. To achieve a higher classification rate, they merge their decisions and soft scores using novel fusion methods which are tested on a wide variety of sequences from the CMU MoCap database. To segment and recognize motion streams, a classification-based approach is proposed in [19] that is based on the extraction of classification feature vectors by utilizing singular value decompositions (SVD) of motion data. The extracted feature vectors capture the dominating geometric structures of motion data as revealed by SVD. Multi-class support vector machine (SVM) classifiers with class probability estimates are explored for classifying the feature vectors in order to segment and recognize motion streams. A more generic approach to motion classification and retrieval of motion capture data is proposed in [19] facilitating the identification of logically related motions scattered in some database. As the main ingredient, they introduce the concept of motion templates (MTs), by which the essence of an entire class of logically related motions can be captured in an explicit and semantically interpretable matrix representation.

3.4 Automatic classification of cyclic activities

In Heritage Crafts and in particular in the context of the Mingei pilots recurring craft processes and techniques are involved. E.g. in the weaving use case three gestures are constantly repeating. In such cases the segmentation of motion can be proven to be a cyclic activity recognition problem. Regarding cyclic activities in [16] a methodology for automatic segmentation and classification of multi-channel data related to motion capture (MoCap) videos of cyclic activities is proposed. A kernel approach is employed to obtain a time representation, which captures the cyclic behaviour of a given multi-channel data.

3.5 Editing and post-processing MoCap animation files

MoCap animation files may vary in quality, accuracy and noise based on the MoCap acquisition technology. This means that in some cases further processing of the data is required in order to be exploitable. This functionality is provided by expensive commercial products and targets the research, gaming and movie industry such as the Autodesk MotionBuilder suite. In the CH sector, though, were end-users are non-experts in the aforementioned technologies these solutions are not only expensive but also require significant technical background. In Mingei we argue that a simple and straightforward application could simplify the task of post-processing, segmenting and annotating motion data and this was the reason for implementing AnimIO which Mingei considers an important addition for the CH sector as described in depth in the following section.

3.6 Progress beyond the state of the art

3.6.1 MoCap datasets

Although allot of datasets are currently available to researchers in order to implement Computer Vision and Machine Learning algorithms there is a lack of such datasets for the case of Heritage Crafts. One of the main contributions of Mingei is that while recording and understanding the crafts in question it is at the same time formulating knowledge in the form of MoCap datasets that will increase the research capacity of the Cultural Heritage sector.

3.6.2 Tools for Cultural Heritage professionals

Mingei is designing and implementing a usable and straightforward authoring tool that will enhance the capacity of the CH sector to exploit MoCap and video recording technologies to create vocabularies of craft processes and techniques. This is valuable resource not only for the scientific exploitation of craft related knowledge by Computer Vision and Motion Capture experts but also for the Cultural Heritage Sector. Using this tool, curators can segment and annotate craft processes and techniques in order to link specific parts of the movement with the tools and materials used.

4. Data

4.1 Data collection

Data acquisition sessions were organised to capture the gestural skills and expertise of each task's experts. Gestures performed under real work conditions were recorded from for the three tasks, and simulated gestures where the expert did not use any materials were recorded also only for the Mastic-cultivation task.

4.1.1 Recording protocol

The recording protocol followed consisted of calibrating the motion capture system, and recording at least three times each task performed by the expert. For the calibration the expert assumed different postures such as I-pose or T-pose, and performed different movements like walking or touching his fingertips, each one for 10 seconds. Before the first recording of each task, the expert was asked to explain each part of the task that he was going to perform, to facilitate the annotation and segmentation of the data afterwards.

4.1.2 Silk-weaving: Recordings at HdS

The recordings took place between April 2nd and 4th, 2019 at the HdS museum, Krefeld, Germany. During that time, one expert was recorded performing the following tasks related to silk weaving: the creation of the punch cards, wrapping of the beam, preparation of the beam, and jacquard weaving with looms of different size (small, medium, and large). These are illustrated in Table 1, and Table 2. Due to the complexity, and long duration of the task, and equipment failure, the wrapping and preparation of the beam were recorded only once.

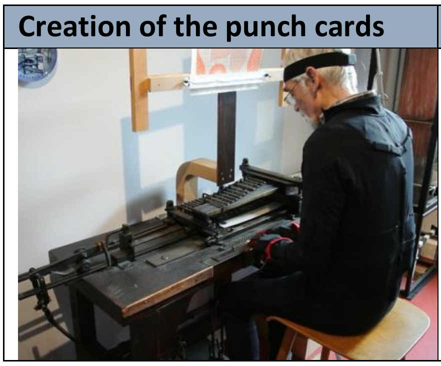
Creation of the punch cards	Wrapping of the beam	Preparation of the beam
		

Table 1. Tasks before the jacquard weaving. (source: Mingei, 2019) [25]

Small size loom	Medium size loom	Large size loom

Table 2. Jacquard weaving with different loom sizes. (source: Mingei, 2019) **[26]**

Only the recordings of jacquard weaving were affected by electromagnetic interference. During these recordings, drifting of the spine, upper arm, and elbow joint angles was detected which was increasing over.

4.1.3 Mastic-cultivation: Recording at Chios Mastic Museum

The cultivation of mastic was recorded in the span of three days, from September 11th to 13th 2019, in Chios, Greece. The recordings of the first and second day were done outside in front of a mastic tree. The recordings of the last day were simulated inside a room. Due to the nature of the cultivation process, each motion was split in different recordings. The tasks related to the cultivation of mastic are illustrated in Table 3, Table 4, and Table 5, and Table 6.

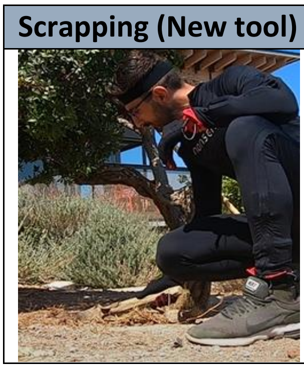
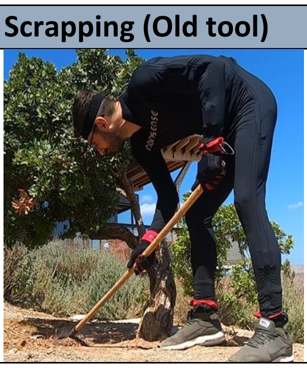
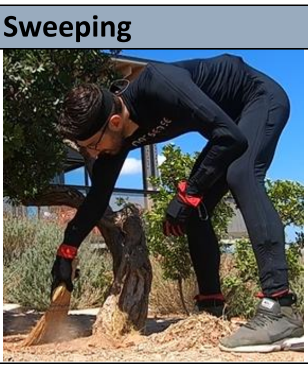
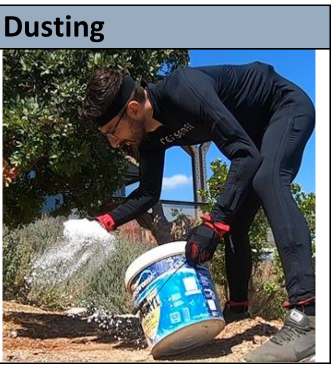
Scrapping (New tool)	Scrapping (Old tool)	Sweeping	Dusting
			

Table 3. Tasks related to the preparation of the tree before the mastic collection. (source: Mingei, 2019) **[27]**

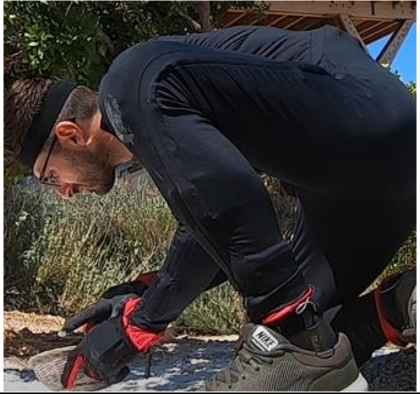
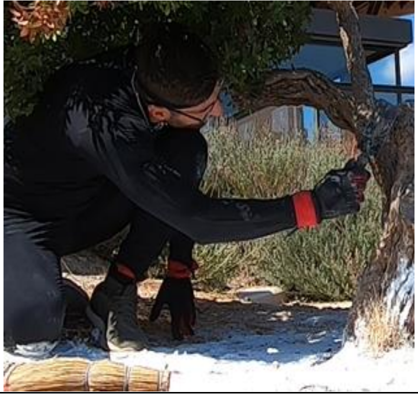
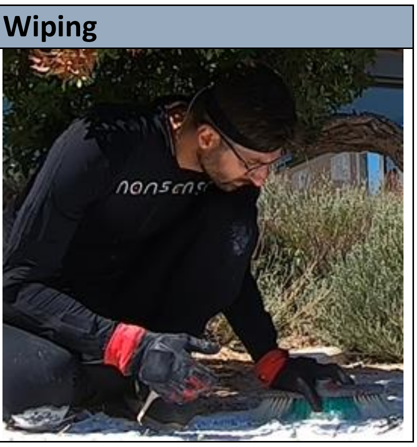
Gathering	Harvesting	Wiping
		

Table 4. Tasks related to the collection of the mastic. (source: Mingei, 2019) [28]

This resulted in separate motion files for each part of the process. In general, the cultivation of mastic was recorded realistically. However, in the actual cultivation and harvesting process, all tasks are days or weeks apart and usually take hours to be completed. As such, the expert had to perform a brief example of the gestures while still being realistic.

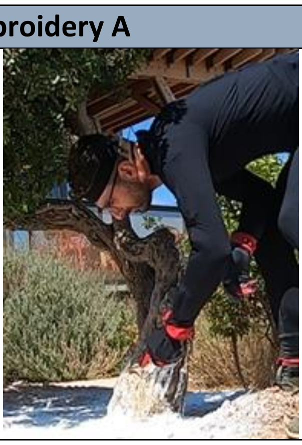
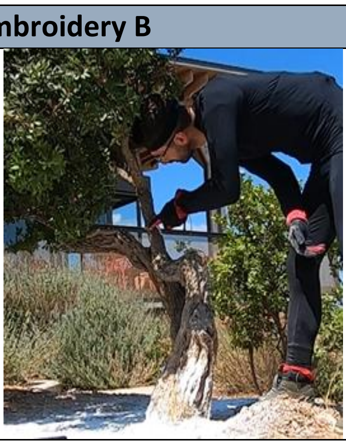
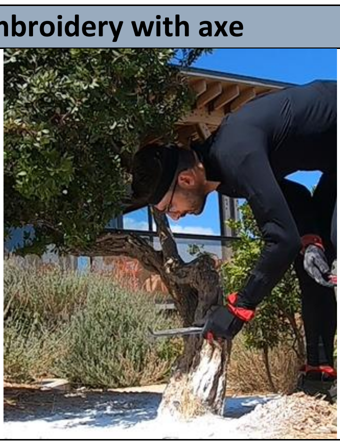
Embroidery A	Embroidery B	Embroidery with axe
		

Table 5. Three different techniques for the task of wounding the tree. (source: Mingei, 2019) [29]

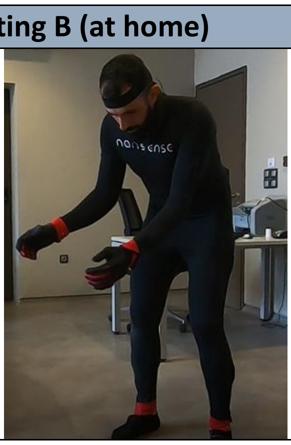
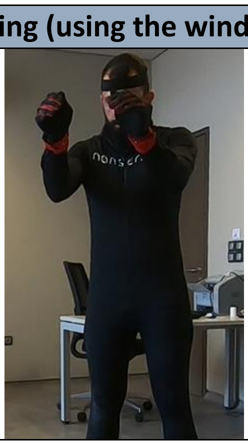
Shifting A (kneeling)	Shifting B (at home)	Cleaning (using the wind)
		

Table 6. Three different ways for the task of cleaning the mastic. (source: Mingei, 2019) [30]

Recordings that required kneeling/sitting on the ground had errors on the legs joint angles. This probably happened because the sensors were interfering with each other. The motion files were corrected offline.

4.1.4 Glass-making: Recording at CERFAV

The creation of a glass jug was recorded on October 11th, 2019 at CERFAV European centre for research and training in the glasswork, Vannes-le-Châtel, France. Because the glass' temperature needed to be consistent through the process, each trial was recorded without any pause in-between gestures. This resulted in one motion file for each attempt which starts from the collection of the molten glass and finishes when the completed jug is being placed to cool. Tables 7, and 8 illustrate the tasks that were recorded for the glass-making use case.

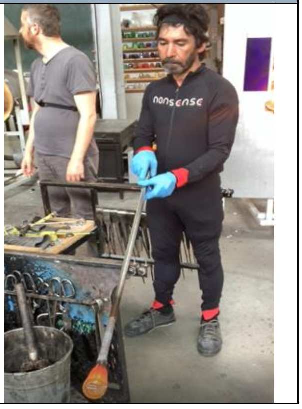
Insert the glass into the furnace	Moving the blowpipe side to side	Shaping glass with the hand	Rotating the blowpipe
			

Table 7. Tasks recorded at CERFAV. (source: Mingei, 2019) [31]

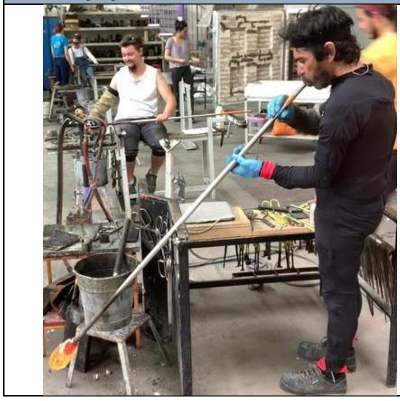
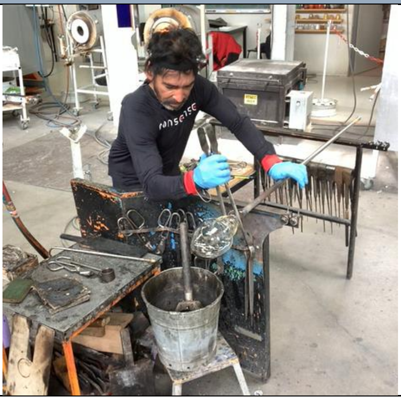
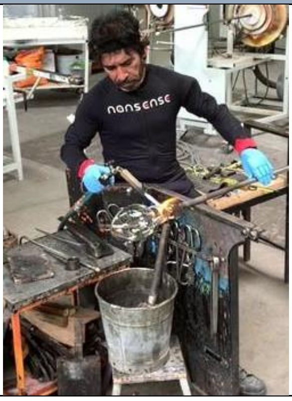
Blowing through the blowpipe	Shaping the glass with the tweezers	Burn the base with the torch
		

Table 8. Tasks recorded at CERFAV. (source: Mingei, 2019) [32]

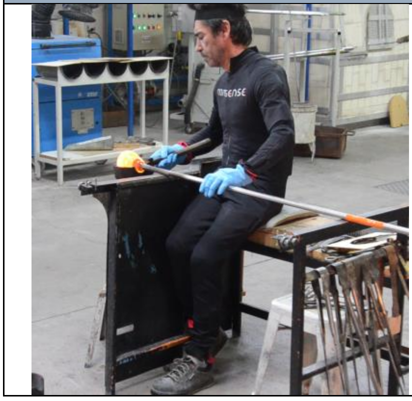
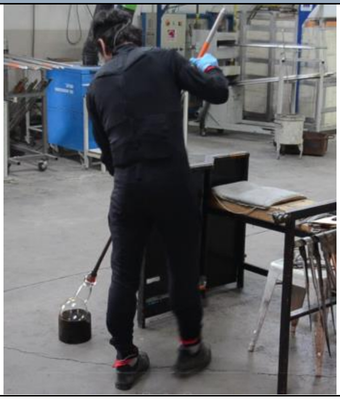
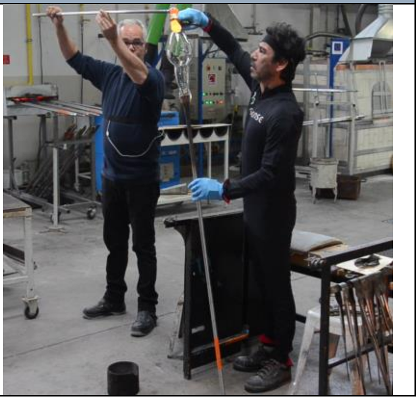
Shaping the glass with the block	Pressing the glass on a metal base	Cutting the glass with the shears
		

Table 9. Tasks recorded at CERFAV. (source: Mingei, 2019) [33]

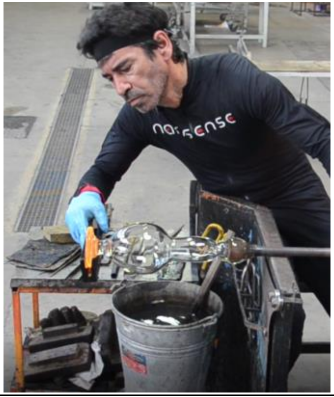
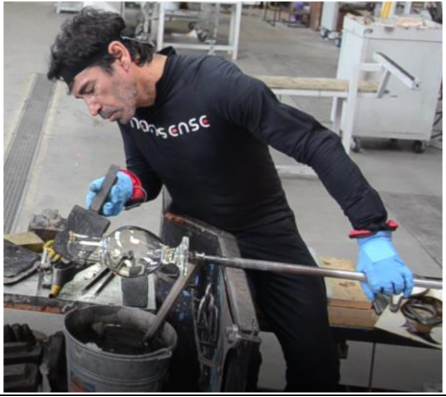
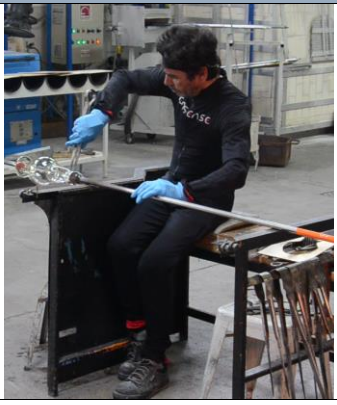
Shaping the base of the jug	Shaping the glass with the paddle	Shaping the glass with the jacks
		

Table 10. Tasks recorded at CERFAV. (source: Mingei, 2019) [34]

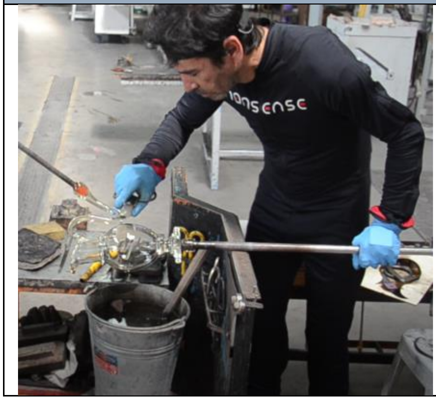
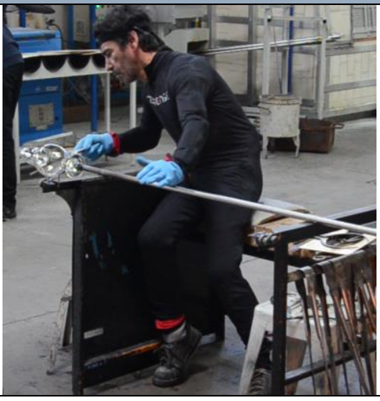
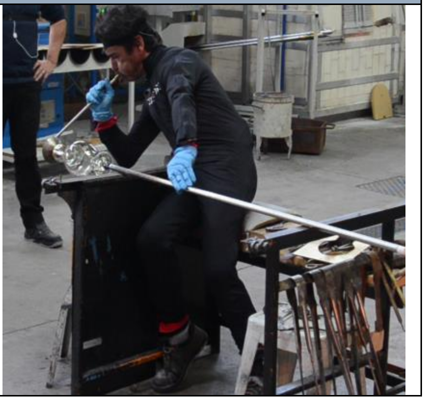
Adding the handle to the jug	Shaping the glass with the tube 1	Blowing through the soffietta
		

Table 11. Tasks recorded at CERFAV. (source: Mingei, 2019) [35]

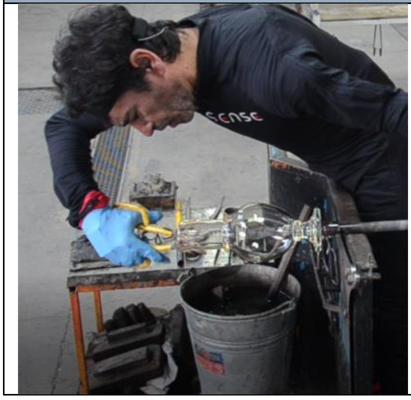
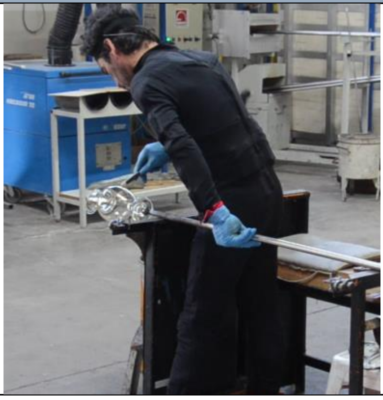
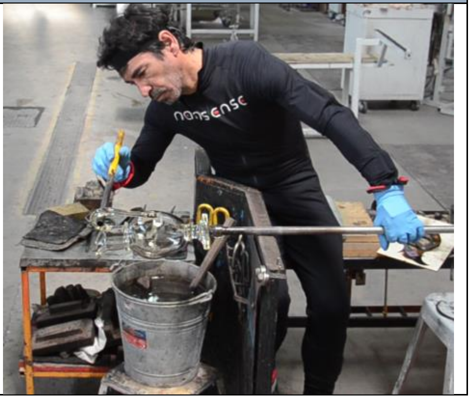
Cutting the jug with the shears	Marking the glass	Shaping the glass with the tube 2
		

Table 12. Tasks recorded at CERFAV. (source: Mingei, 2019) [36]

The creation of a glass jug had three steps. The first step was to inflate and shape the molten glass in the jug's main body (container). The next step was to create the base, and finally the handle. Throughout the task, the expert was spinning and shaping the jug to prevent the glass deforming due to gravity. Lastly, an assistant was also required to blow into the glass while the expert was shaping the main body. For the sake of completeness, the breathing of the assistant was also recorded, as shown in Figure 3.

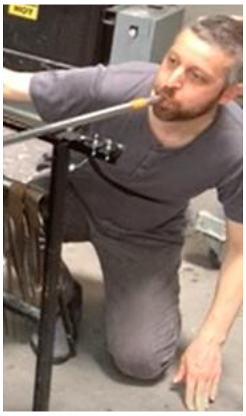


Figure 1. Breathing Recording of Glass Blowing. (source: Mingei, 2019) [20]

In general, the recordings were without any serious interference. However, it was difficult for the expert to manipulate objects with the gloves of the suit. To solve that, latex gloves were worn on top of the suit's lycra gloves to add more friction to the fingers.

4.2 Data processing

The processing of the MoCap consisted of two steps, first the application of a low pass filter, and then the correction of erroneous postures either caused by electromagnetic interference or sensors drifting when the task was going on for too long and another calibration was required.

4.2.1 Filtering

Initially, a low pass Butterworth filter was applied to the raw MoCap data to eliminate high-frequency noise. The cut-off frequency was determined according to the power spectrum density of the signal, in order not to over smooth the signal. In Figure 4, it can be seen the noise that was removed from a segment of the signal that represents the right forearm flexion/extension. Figure 5 shows the joint angles from the right arm before and after the noise removal.

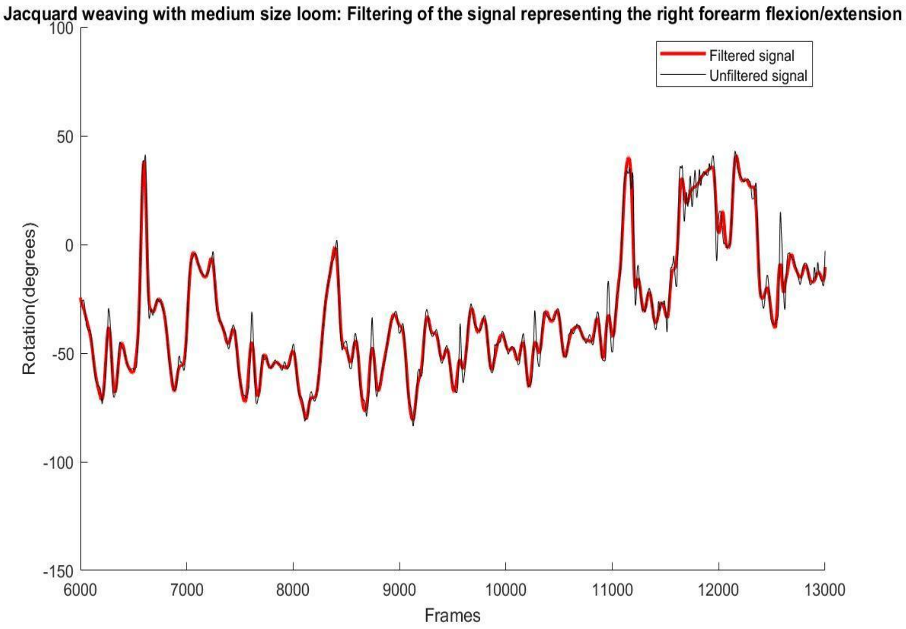


Figure 2. Low pass filtering of the right forearm joint angles that indicate the flexion/extension. (source: Mingei, 2019) [21]

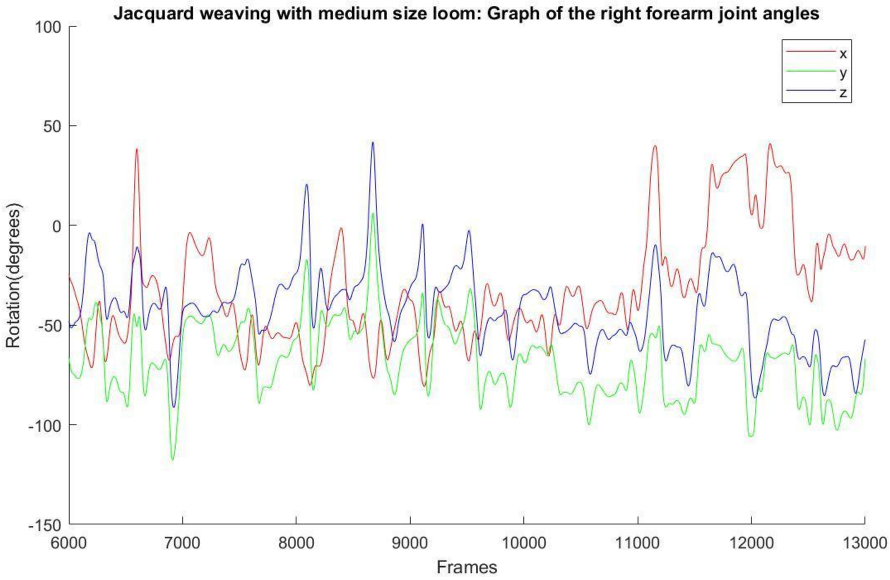


Figure 3. Right forearm joint angles processed: 'x' corresponds to flexion/rotation, 'y' to supination/pronation, and 'z' to carrying angle. (source: Mingei, 2019) [22]

4.2.2 Correction of erroneous postures

During the recordings, the MoCap system the sensors could drift or be affected by magnetic disturbances from nearby metallic objects. As a result, occasionally, unrealistic joint angles were recorded in otherwise accurate motion capture. To correct this error, the recordings were edited using a 3D character animation software (MotionBuilder, Autodesk Inc., San Rafael, CA. USA). Using the software, the unrealistic motions were corrected based on common sense and video feedback. Figure 6 shows an example of a posture correction from the HdS recordings.

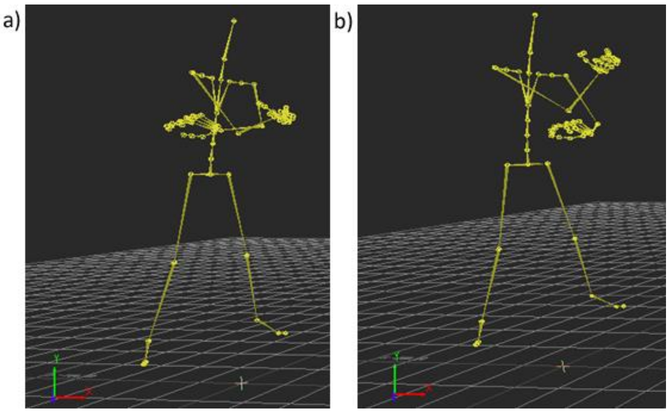


Figure 4.Posture correction with MotionBuilder: a) Erroneous posture, b) Posture corrected. (source: Mingei, 2019) [23]

4.2.3 Data segmentation

After the correction and noise removal from the MoCap data, the data was segmented by gestures. The recordings initially were done by tasks, being one recording a whole task, these recordings were afterwards segmented by gestures. One task could have only one gesture that is repeated several times or might have more gestures that are repeated through the task. For example, jacquard weaving consisted of the sequential repetition of three gestures. Tables 9, 10, and 11 illustrate the number of gestures extracted from each task, how many repetitions were recorded from each, and its mean duration. In general, the goal was to collect a minimum of three repetitions per gestures. However, for a few cases, it wasn’t possible to get that number. More specifically, for the “beam preparation” task, the equipment broke, and it wasn’t possible to repeat the task. For the “cutting the glass with the shears” the recorded data were heavily distorted. Though it is possible to restore them to some extent, the outcome will be interpolated data, rather than true recordings. However, the need for three repetitions per gesture was established for redundancy because it is a general practice to expect a portion of MoCap recordings done outdoors to be unusable. Strictly speaking, only one repetition per task is required for the scope of Mingei with any additional recording to serve as additional information or backup. Though, it would have been ideal to have a few more recordings for certain gestures, the current dataset will not impact the expected outcome in any detrimental way.

Performer	Task	Gestures	Repetitions	Mean duration
1	Creating a card	Gesture 1	110	5.55
	Beam preparation	Gesture 1	3	34.14
		Gesture 2	2	58.39
		Gesture 3	4	79.25
		Gesture 4	1	8.97
		Gesture 5	1	323.03
	Wrapping the beam	Gesture 1	124	4.89
	Weaving with small size loom	Gesture 1	11	2.38
		Gesture 2	11	5.33
		Gesture 3	11	4.04
	Weaving with medium size loom	Gesture 1	35	1.16
		Gesture 2	35	2.07
		Gesture 3	35	2.20
	Weaving with large size loom	Gesture 1	16	0.75
		Gesture 2	16	1.18
		Gesture 3	15	2.37

Table 13. Repetitions and mean durations of gestures extracted from the tasks recorded at HdS. (source: Mingei, 2019) [37]

Performer	Task	Gestures	Repetitions	Mean duration
1	Scrapping (New tool)	Gesture 1	3	7.29
	Scrapping (Old tool)	Gesture 1	6	16.36
	Sweeping	Gesture 1	6	10.06
	Dusting	Gesture 1	6	11.47
	Embroidery A	Gesture 1	6	9.45
	Embroidery B	Gesture 1	3	6.91
	Embroidery with Axe	Gesture 1	3	4.96
	Gathering	Gesture 1	5	12.57
	Harvesting	Gesture 1	3	8.02
	Wiping	Gesture 1	3	8.85
	Shifting A	Gesture 1	3	9.36
2	Scrapping (Old tool)	Gesture 1	3	4.53
	Sweeping	Gesture 1	3	3.87
	Dusting	Gesture 1	3	4.03
	Embroidery A	Gesture 1	3	5.12
	Gathering	Gesture 1	3	4.08
	Harvesting	Gesture 1	4	4.34
	Wiping	Gesture 1	3	3.18
	Shifting A	Gesture 1	3	6.41
	Shifting B	Gesture 1	3	7.40

	Cleaning (using the wind)	Gesture 1	3	3.49

Table 14. Repetitions and mean durations of gestures extracted from the tasks recorded at Chios Mastic Museum. (source: Mingei, 2019) [38]

Performer	Task	Gestures	Repetitions	Mean duration
1	Inserting the glass into the furnace	Gesture 1	8	9.42
	Moving the blowpipe side to side	Gesture 1	85	25.92
	Shaping the glass with the hand	Gesture 1	2	2.92
	Rotating the blowpipe	Gesture 1	40	7.49
	Blowing through the blowpipe	Gesture 1	17	16.83
	Shaping the glass with the tweezers	Gesture 1	20	2.23
		Gesture 2	36	13.20
	Burn the base with the torch	Gesture 1	19	4.22
	Shaping the glass with the block	Gesture 1	4	3.93
	Pressing the glass on a metal base	Gesture 1	6	6.70
	Cutting the glass with the shears	Gesture 1	1	4.04
	Shaping the base of the jug	Gesture 1	8	7.06
	Shaping the glass with the paddle	Gesture 1	6	6.74
	Shaping the glass with the jacks	Gesture 1	7	16.66
	Adding the handle to the jug	Gesture 1	15	8.37
	Shaping the glass with the tube 1	Gesture 1	3	9.84
	Blowing through the soffietta	Gesture 1	4	26.00
	Cutting the jug with the shears	Gesture 1	10	19.63
	Marking the glass	Gesture 1	6	9.37

Table 15. Repetitions and mean durations of gestures extracted from the tasks recorded at CERFAV. (source: Mingei, 2019) [39]

4.3 Data deliverables

The processed and segmented MoCap data is delivered in the standard BVH format, and come along with an XML file containing the metadata of the BVH file.

4.3.1 Files naming

The BVH files and XML files are provided with the same naming nomenclature, structured as follows:

[Usecase][Task][Location][Date][P#][G#][R#](.bvh/.xml)

Where *Use case* could be Silk, Glass or Mastic. *Task* is the name of the task that the expert was doing, for example weaving, gathering or blowing. *Location* is either Hds, PIOP (for the Chios Mastic Museum) or CERFAV. *Date* indicates the date of the recording, with the format YYYYMMDD. *P#* is to indicate the expert that performed the task recorded, for example P1 or P2 (performer 1 or performer 2). *G#* corresponds to the gesture contained in the MoCap file (G1, G2, G3, etc.), and finally, *R#* is the number of repetition of the gesture (R1, R2, R3, etc.).

4.3.2 BVH files

The resulting processed and segmented MoCap data is provided in BVH format. This file format was designed for motion capture data exchange. It is divided in two parts: **Hierarchy** segment and **Motion segment**.

Hierarchy is the section where the skeleton hierarchy is described, in a depth-first tree like fashion. The tree's nodes are the skeleton joints. For each joint, the information includes an offset from the parent joint in world space units and a sequence of the channels that will be provided in the Motion section for this joint(per-frame). The skeletal structure has a root site, a joint site, and an end site. Root is the root node of the tree. Manipulating the root joint, means affecting the whole skeleton. Joint is used for all intermediate joints and the end site is used for the ending joints. Offsets are specified for a right handed coordinate system, aligned with the world. By using the calculated position of a joint in world space and the offset of one of its children, we can calculate the length and direction of the equivalent bone. The end site is a pseudojoint that exists only to enable the aforementioned direction for the leaf joints. As such, Root and Joint entries define a joint name, whereas End Site does not need one. The second part of the joint node is defined as Channels, is the count and sequence of attributes which define the order with which the motion data should be read. Usually only the root joint contains translation attributes. The rest of the Joints (excluding End Sites) contain mainly the 3 rotational attributes and the order of the attributes in the sequence defines the order by which rotations will be applied to the joints. For example, the line CHANNELS 3 Z-rotation Y-rotation X-rotation means that this joint has rotation data in each frame where the rotation around the local Z axis is applied first, then the rotation around the local Y axis and finally the rotation around the local X axis. The rotation data are provided in Euler angle form and correspond to Axis Aligned - Axis Angle rotations. Basically the rotations are angles applied on the joints, where the local coordinate frame is aligned with the world's.

The **Motion** segment contains the recorded joint angles. First, we get metadata for the animation data like the number of frames (frame count) in the animation, as well as, the interval between the frames in seconds (frame interval). Thus the duration of the animation can be calculated by multiplication of the frame count and the frame interval. Each line provided has the translation (where applicable) and the joint angles of each frame of the animation in order of definition(depth-first). In essence, the hierarchy segment defines the order which each value corresponds to a specific translation or rotation.

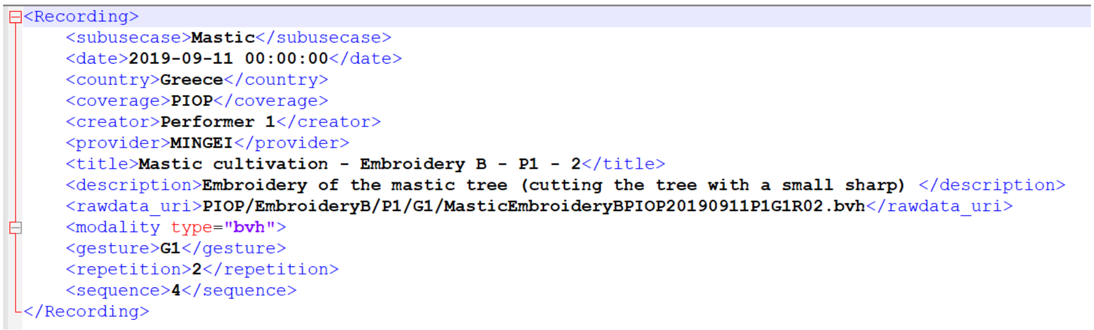
The BVH files that were recorded have motion data with a sample rate of 90 frames per second, and correspond to the x, y, and z local rotations of 25 body segments, rotations from 36 fingers segments, and 4 from two toes with the pelvis having 3 additional degrees of freedom that represent the translation of the whole body.

4.3.3 XML files

The XML files were added to the data deliverables to provide the metadata of the motion files. The content and description of each section of the XML files are shown in Table 12. A screenshot from an XML file is illustrated as an example in Figure 7.

Section	Description
Subusecase	Name of the pilot
Date	Date of the recording
Country	Country where the recording was carried out
Coverage	Institution were the recording was carried out
Creator	The performer of the task
Provider	The name of the project
Title	Title of the file
Description	Description of the task recorded
Uri	Path to the corresponding BHV file
Modality	MoCap file format
Gesture	Gesture contained in the BVH file
Repetition	Number of repetition of the gesture
Sequence	Sequence number that indicates order of the gestures

Table 16. Content description of the XML files. (source: Mingei, 2019) [40]



```
<Recording>
    <subusecase>Mastic</subusecase>
    <date>2019-09-11 00:00:00</date>
    <country>Greece</country>
    <coverage>PIOP</coverage>
    <creator>Performer 1</creator>
    <provider>MINGEI</provider>
    <title>Mastic cultivation - Embroidery B - P1 - 2</title>
    <description>Embroidery of the mastic tree (cutting the tree with a small sharp) </description>
    <rawdata_uri>PIOP/EmbroideryB/P1/G1/MasticEmbroideryBPIOP20190911P1G1R02.bvh</rawdata_uri>
    <modality type="bvh">
    <gesture>G1</gesture>
    <repetition>2</repetition>
    <sequence>4</sequence>
</Recording>
```

Figure 5.Example of an XML file. (source: Mingei, 2019) [24]

5. Animation Studio (AnimIO)

AnimIO was created to address important gaps in the state of the art regarding the segmentation of MoCap and video data as summarised below:

(a) MoCap editing software are to be used by experts, scientist and computer animators
(b) Video editing software are to be used by video editing experts
(c) There is no software that supports synchronised MoCap and Video editing

In Mingei, post-processing of MoCap animation files is important in order to generate a vocabulary of gestures. This is an essential part of the process as in the case of HCs allot of repetition exists in the MoCap data and users should be able to isolate parts of the movement that represent important actions of the craft. For editing MoCap animation files, Mingei is implementing a BHV editor, called Animation Studio (AnimIO). AnimIO allows visualization, editing, and annotation of 3D animation files, obtained by motion capture or visual tracking. In the case of visual tracking, temporally corresponding video can be also edited. The application allows the user to isolate animation segments and the associated video for further annotation, as well as the synthesis of composite animation files and videos from such segments. Pertinent annotation software exists in the linguistics domain [1], but does not stream for video and motion capture. Using AnimIO, segments of the BVH input can be isolated and exported to test different "atomic" scenarios (e.g. hand movement when hammering a nail), simplify the input, or to allow the in depth analysis of certain scenarios.

Rational and relation to Mingei's vision: Isolated parts of the craft will be connected to the CrO to represent schemas regarding how a craft is practiced. These schemas will be used to implement motion driven narratives that interweave historical and societal information with demonstrations of craft practice in a virtual environment. In such a context, the narrator (a virtual human avatar in the 3D world) will employ annotated video segments to accompany the narration while the virtual practitioner will replay motions based on the isolated gestures of the craft.

5.1 Functionality

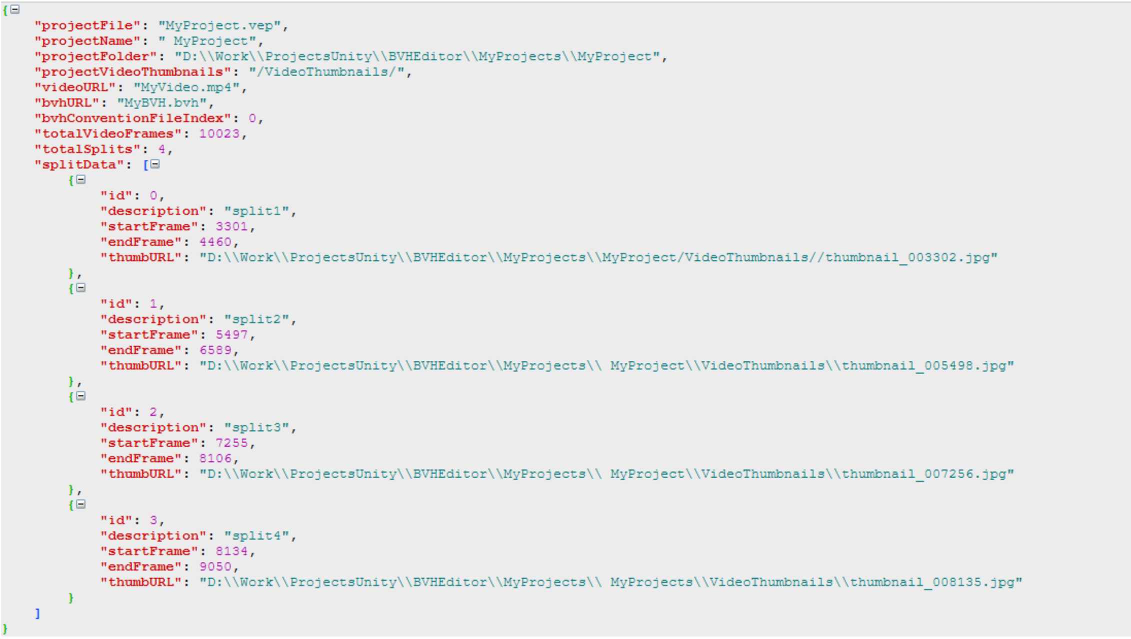
5.1.1 AnimIO project structure

AnimIO supports the creation of motion editing projects which are encoded with the ".vep" file suffix. This file contains all the required information needed by the editor which is, the project file name (**projectFile**), project name (**projectName**), project folder (**projectFolder**), the video (**videoURL**) and BVH (**bvhURL**) urls, as well as the BVH convention file index (**bvhConventionFileIndex**). All of the above parameters must be provided by the user at the creation of the project with the exception of video and BVH urls where the user has the capability to provide only one of them (a project can contain a video, a BVH or both).

A ".vep" file is automatically generated by AnimIO based on user selections and no editing is required for the user's file directly on the .vep file.

Additional parameters are being created by the editor but the user does not have direct access to them. Those parameters are the **totalVideoFrames**, which stores the number of frames for the video or the BVH file (video files always takes priority, if the project contains both file formats), the number of segments the user has created (video/BVH parts), and finally the information about those segments. Each segment is defined by an id, a description, a start frame (**startFrame**), an end frame (**endFrame**), and a thumbnail url (**thumbURL**) which is usually the first frame image of the selected part of the video or BVH.

AnimIO project files are encode into json string as presented in the Figure 6. The following figure presents the internal structure of the ".vep" file generated by AnimIO and is not directly editable by end users.



```
{⊟
    "projectFile": "MyProject.vep",
    "projectName": " MyProject",
    "projectFolder": "D:\\Work\\ProjectsUnity\\BVHEditor\\MyProjects\\MyProject",
    "projectVideoThumbnails": "/VideoThumbnails/",
    "videoURL": "MyVideo.mp4",
    "bvhURL": "MyBVH.bvh",
    "bvhConventionFileIndex": 0,
    "totalVideoFrames": 10023,
    "totalSplits": 4,
    "splitData": [⊟
        {⊟
            "id": 0,
            "description": "split1",
            "startFrame": 3301,
            "endFrame": 4460,
            "thumbURL": "D:\\Work\\ProjectsUnity\\BVHEditor\\MyProjects\\MyProject/VideoThumbnails//thumbnail_003302.jpg"
        },
        {⊟
            "id": 1,
            "description": "split2",
            "startFrame": 5497,
            "endFrame": 6589,
            "thumbURL": "D:\\Work\\ProjectsUnity\\BVHEditor\\MyProjects\\ MyProject\\VideoThumbnails\\thumbnail_005498.jpg"
        },
        {⊟
            "id": 2,
            "description": "split3",
            "startFrame": 7255,
            "endFrame": 8106,
            "thumbURL": "D:\\Work\\ProjectsUnity\\BVHEditor\\MyProjects\\ MyProject\\VideoThumbnails\\thumbnail_007256.jpg"
        },
        {⊟
            "id": 3,
            "description": "split4",
            "startFrame": 8134,
            "endFrame": 9050,
            "thumbURL": "D:\\Work\\ProjectsUnity\\BVHEditor\\MyProjects\\ MyProjects\\VideoThumbnails\\thumbnail_008135.jpg"
        }
    ]
}
```

Figure 6.Example of an AnimIO project file. (source: Mingei, 2019) [43]

5.1.2 Tool structure

AnimIO editor is structured in a way similar to video processing software in order to make it more intuitive and user friendly. To this end, when the software is launched the starting interface is divided in four sections (see Figure 7):

- **1 - Project tool panel:** From this panel, users can create, open, save, and export AnimIO projects
- **2 - Video playback panel:** This panel displays the video file that is currently linked with the active **AnimIO** project
- **3 - BVH playback panel:** This panel displays the skeleton-based BVH file that is currently linked with the active **AnimIO** project
- **4 - Video-BVH playback timeline:** This is the timeline panel where users can control playback and create segments of the video and BVH files using the same interaction metaphor employed by most video editing software.

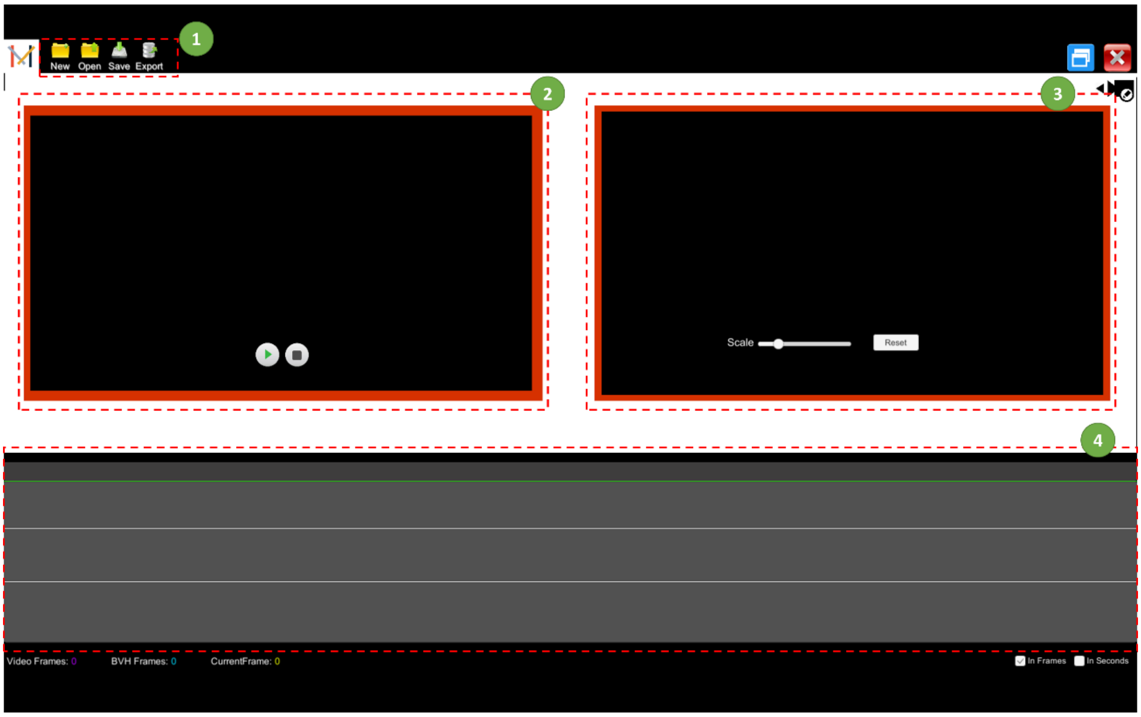


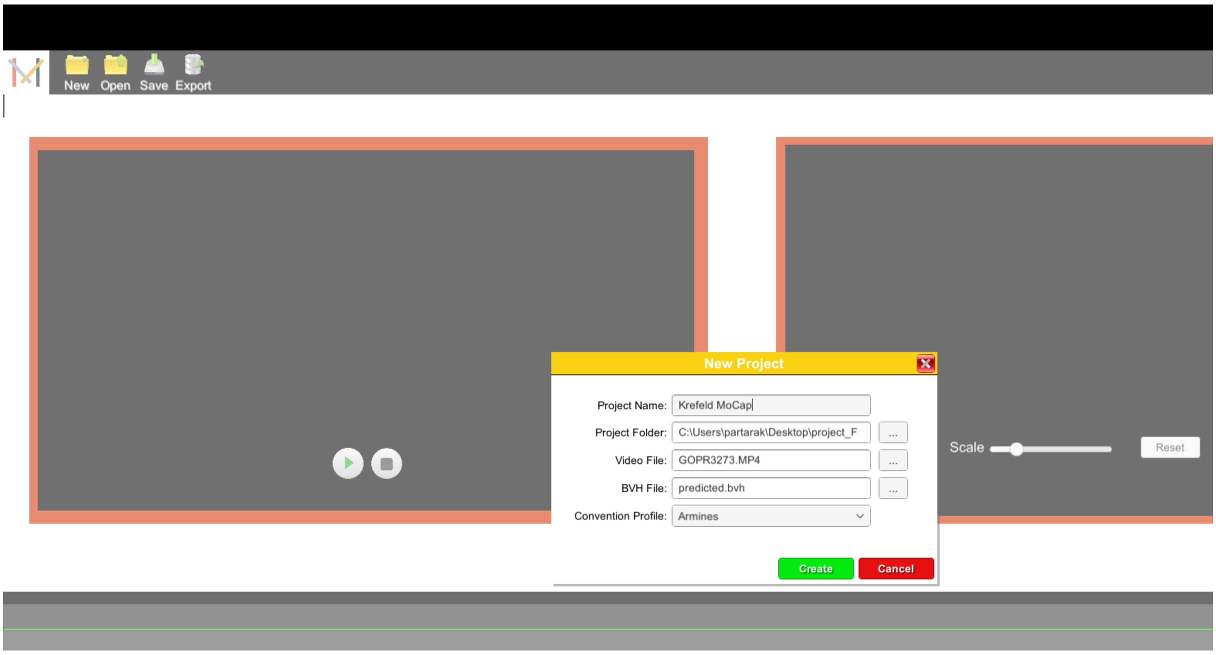
Figure 7. AnimIO tool structure. (source: Mingei, 2019) [43]

5.1.3 Main functionality

In this sub-section an overview of the main functions of the AnimIO is presented.

5.1.3.1 Create AnimIO Project

To create an AnimIO project, users should execute "AnimIO.exe" and then locate the "new" button icon at the top left side of the main screen. A modal dialog is presented where users should enter the details for the project: **Project Name**, **Project Folder**, **BVH source**, **Video source** and **conversion profile** (see Figure 8). The skeleton formats used by FORTH and ARMINES are converted to ensure that the editor will work with both conversions. Not all fields are mandatory though. For example, a project may have only BVH and not video source.



New Project

Project Name: Krefeld MoCap

Project Folder: C:\Users\partarak\Desktop\project_F

Video File: GOPR3273.MP4

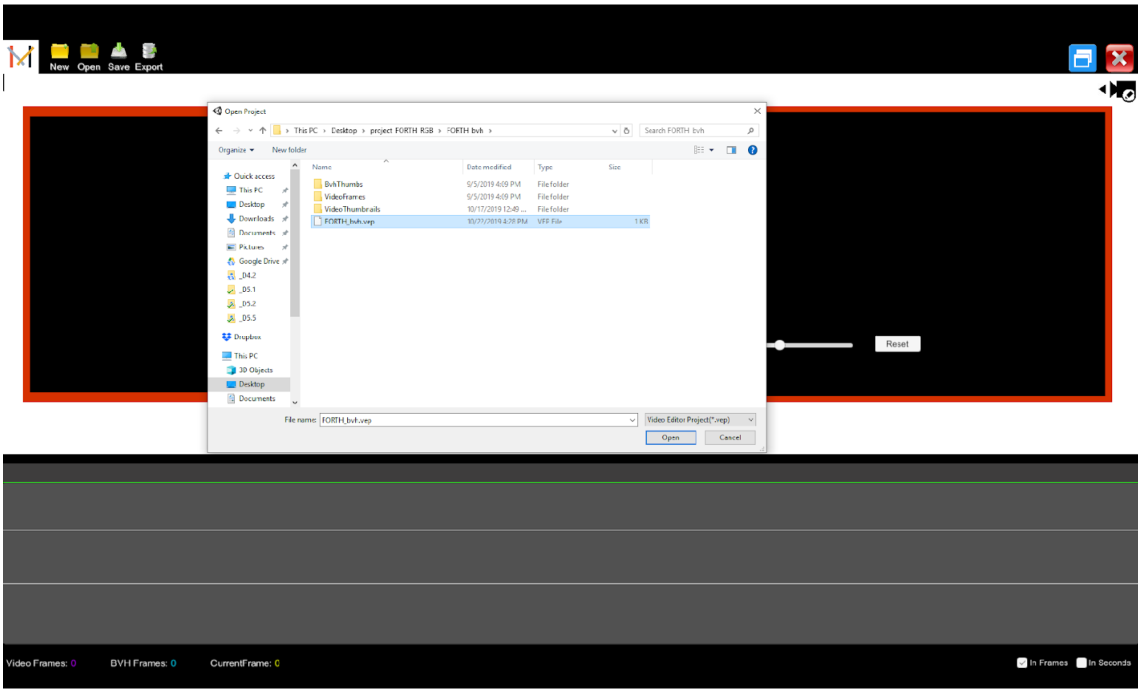
BVH File: predicted.bvh

Convention Profile: Armines

Figure 8. Create AnimIO project. (source: Mingei, 2019) [43]

5.1.3.2 Load project

To load an existing AnimIO project users should execute the "AnimIO.exe" and use the open button icon at the top left side of the main screen. Then, a file explorer appears where users are requested to locate and select the ".vep" project file (see Figure 9).



Name	Date modified	Type	Size
BvhThumbs	9/5/2019 4:09 PM	File folder	
VideoFrames	9/5/2019 4:09 PM	File folder	
VideoThumbnails	10/17/2019 12:49 ...	File folder	
FORTH_bvh.vep	10/22/2019 4:28 PM	VEP File	1 KB

File name: FORTH_bvh.vep

Video Editor Project(*.vep)

Video Frames: 0 BVH Frames: 0 CurrentFrame: 0

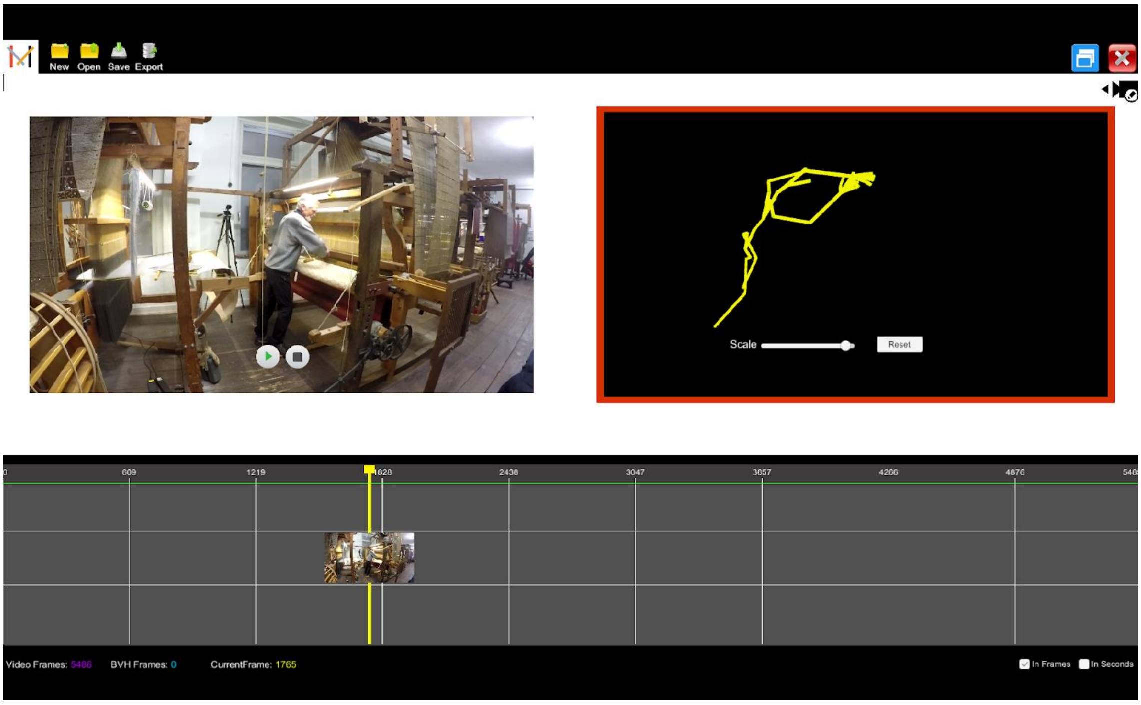
Figure 9. Load project. (source: Mingei, 2019) [43]

5.1.3.3 Save AnimIO Project

At any point, the AnimIO tool can save the current project. To do that, use the save button at the top left side of the screen. This operation saves all progress to the currently loaded AnimIO project.

5.1.3.4 Synchronised playback of video and BVH

In the case where the project has both BVH and video source synchronized playback is the process where users can reproduce the video and the 3D representation of the extracted movement side by side. This is simply done by selecting the play button of the video section of the AnimIO. The result is the initiation of the synchronized playback. To stop the playback users can just select the stop button of the video section (see Figure 10)



Video Frames: 5486 BVH Frames: 0 CurrentFrame: 1765

☑ In Frames □ In Seconds

Figure 10. Synchronized playback. (source: Mingei, 2019) [43]

5.1.3.5 Rotate, Translate and Scale skeleton view

The 3D representation of the extracted movement appears on the left side of the AnimIO window. From this window, users can rotate, translate and scale the skeleton position (see Figure 11):

- **Translation** is done by pressing the left mouse button and dragging across the BVH preview window.
- **Rotation** is done by pressing the right mouse button and dragging across the BVH preview window.
- **Scaling** is done using the scale slider
- **Reset** to the default values can be done at any time of playback by selecting the reset button

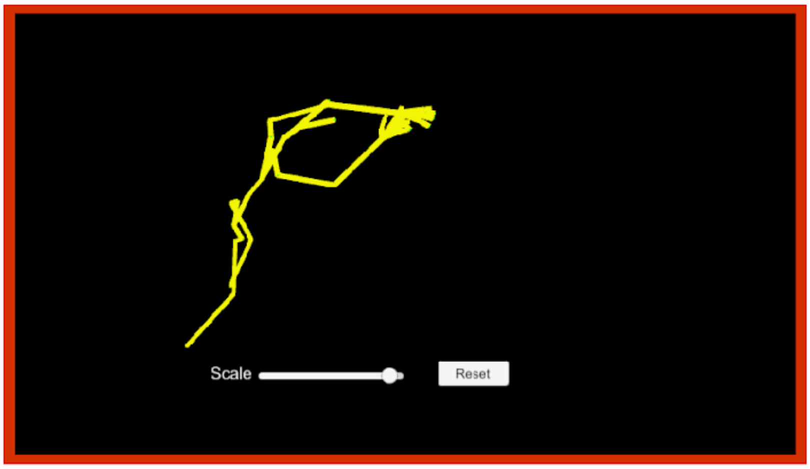


Figure 11. Adjust skeleton view. (source: Mingei, 2019) [43]

5.1.3.6 Skeleton appearance settings

Currently AnimIO studio is previewing motion in a simplified stick motion format. This format visualises abstractly the motion by displaying the rotation and translation of skeleton joints in space linked with simple line segments. This visualisation is the absolute baseline scientific representation of motion and it was selected in the initial versions of the application for several technical reasons/considerations:

- Mingei facilitates MoCap data from alternative data sources which are: (a) Nansense suit, (b) Full body visual tracking and (c) hands visual tracking. Although all these sources are using the same data standard (BVH) there are different settings in terms of scaling, translation, world coordinate system in each of the data source. Source filters have been implemented for the harmonisation of different data sources so as to be translated in the same system and thus being rendered gracefully.
- Mingei currently supports only generic interventions to the visualised data sources by allowing the selection of different simplified animation elements such as cubes, spheres, etc. that enhance visualisation. A more sophisticated approach will require further work covered in the following bullet point
- AnimIO is previewing animation files on the fly and currently does not retarget animation files to skinned avatar models which will provide a more readable representation for CH professionals. This is part of the work currently conducted in the tool and regards the implementation of skinned avatars for each different BVH profile employed by the project.

Of course the representation of BVH animation files in the context of experiences for end users is covered through a different process, workflow and Mingei tools focusing on motion retargeting, motion visualisation and interaction with tools and machines. This part of Mingei work is presented in depth in D5.2

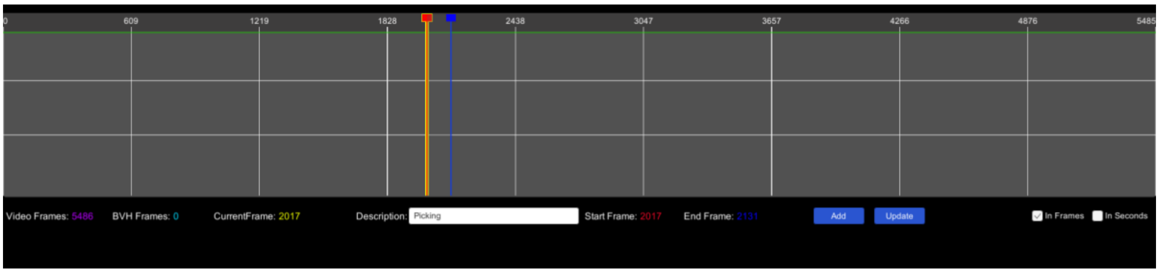
5.1.3.7 Skeleton focusing settings

During the design of AnimIO several comments of the usability experts reviewing the system were targeted to the usability of the skeleton previewing functionality. In this context several iterations of design were made in order to minimise issues of interacting (scaling, translating, rotating) the skeletal view. These requests were related to the possibility of inexperienced to 3D technologies CH users being able to manipulate the skeleton preview in order to understand the mapping of video to MoCap. This was even more challenging when only MoCap is loaded where no video alternative is present. Although many adjustments were made, more are required.

Mingei is considering to provide viewing pre-sets on the skeleton preview panel to allow end users to easily focus on several parts of the previewed bvh such as auto scale to skeleton bounding box, upper body close up, lower body close up, hands close up, etc.

5.1.3.8 Create Segment of the Video and BVH

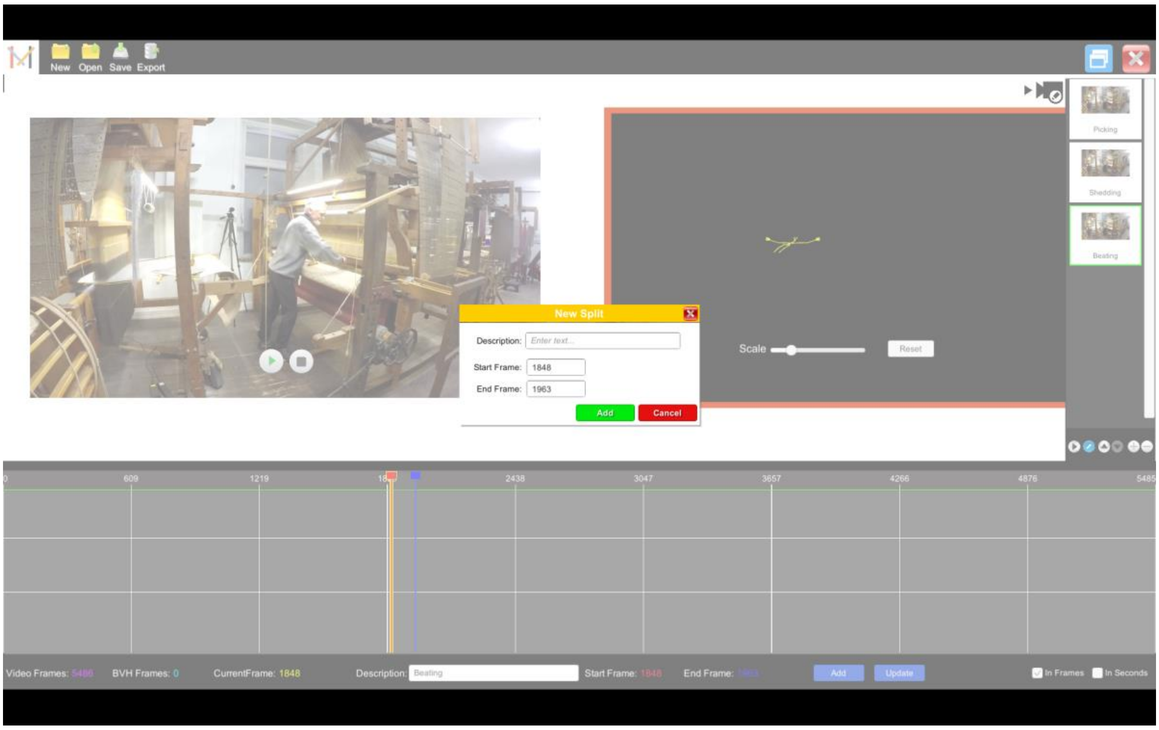
To create a segment user should perform a double click on the video timeline at the bottom side of the main window. Two vertical lines will appear, one coloured red marking the beginning of the segment and a blue-coloured line to mark the end. Users should drag the lines to mark the start and end point of the segment (see Figure 12). Right click and select the 'create new segment' menu option.



Video Frames: 5486 BVH Frames: 0 CurrentFrame: 2017 Description: Picking Start Frame: 2017 End Frame: 2131

Figure 12. **Creating a segment (red-line: segment start, blue-line: segment end). (source: Mingei, 2019) [43]**

A pop-up window will appear for the name of the annotated segment (see Figure 13). The newly created segment will be added to the list of segments.



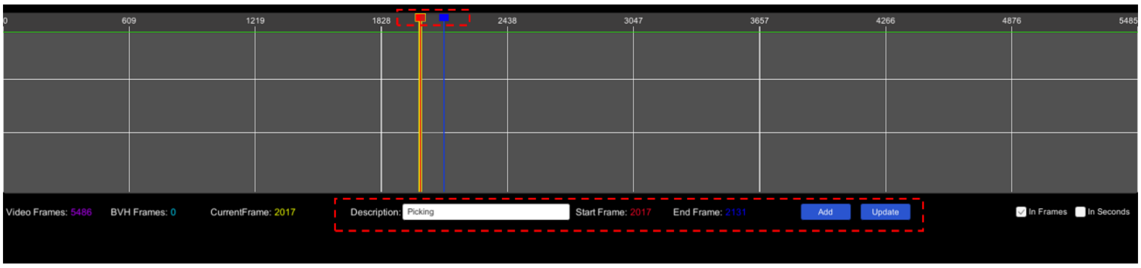
Description: Enter text...
Start Frame: 1848
End Frame: 1963

Video Frames: 5486 BVH Frames: 0 CurrentFrame: 1848 Description: Beating Start Frame: 1848 End Frame: 1963

Figure 13. Create Segment of the Video and BVH. (source: Mingei, 2019) [43]

5.1.3.9 Update segment

To update a segment, the users should first select the segment from the list of segments and alter the starting and/or ending frames by dragging the vertical lines on the timeline. After the new starting and/or ending points are selected, use the update button to save the changes (see Figure 14).

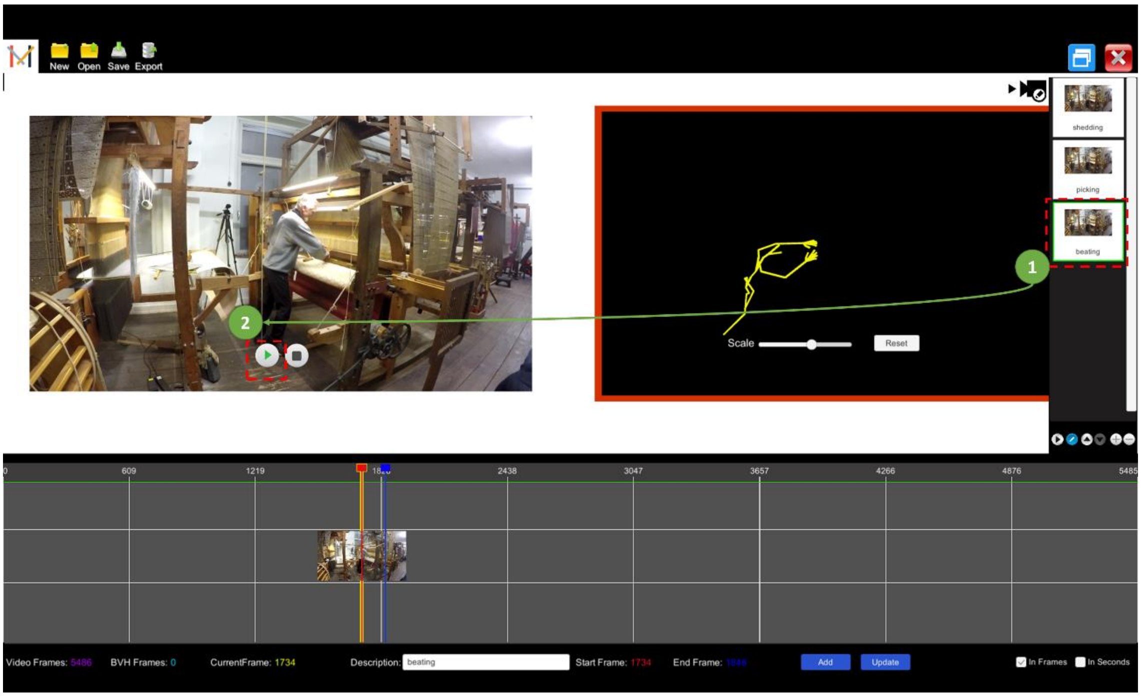


Video Frames: 5486 BVH Frames: 0 CurrentFrame: 2017 Description: Picking Start Frame: 2017 End Frame: 2131

Figure 14. Update segment. (source: Mingei, 2019) [43]

5.1.3.10 Playback segment

To replay a video segment it needs to be selected from the list of available segments and then use the playback function (see Figure 15).



Video Frames: 5486 BVH Frames: 0 CurrentFrame: 1734 Description: beating Start Frame: 1734 End Frame: [illegible] Add Update ☑ In Frames □ In Seconds

Figure 15. Playback segment. (source: Mingei, 2019) [43]

5.1.3.11 Reorder segments

Using the list of segments the annotated parts can be reordered. To change the order, the directional arrows at the bottom side of the list can be used to move the selected segment (see Figure 16).

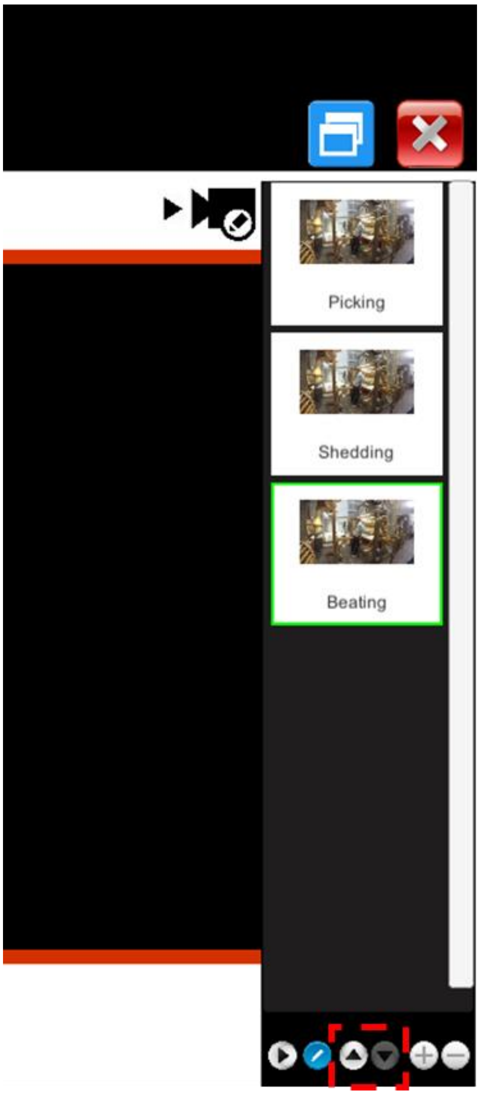


Figure 16. Reorder segments. (source: Mingei, 2019) [43]

5.1.3.12 Remove segment

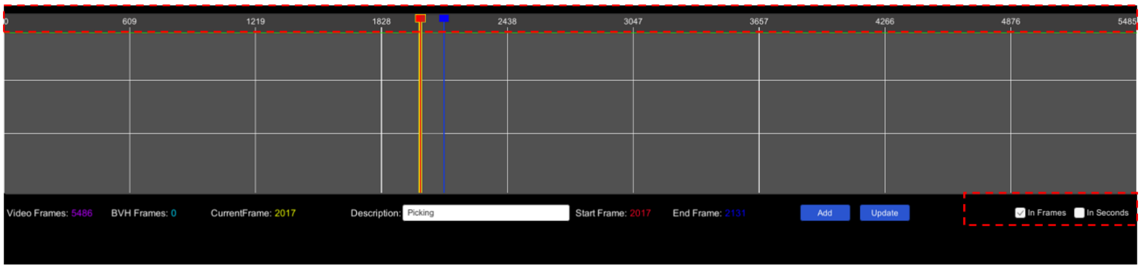
To remove a segment, use remove button that is located underneath the segment list on the selected part.

5.1.3.13 Create new segment manually

To manually create a new segment, users should right click the timeline control and then select the “new segment”. A new popup dialog will appear to fill-in the segment title, the start, and the end frame. Selects the “add” button to complete the process.

5.1.3.14 Select playback timing modality

In order to change the playback modality users should locate and check the “seconds” or “frames” tick box under the timeline bar.



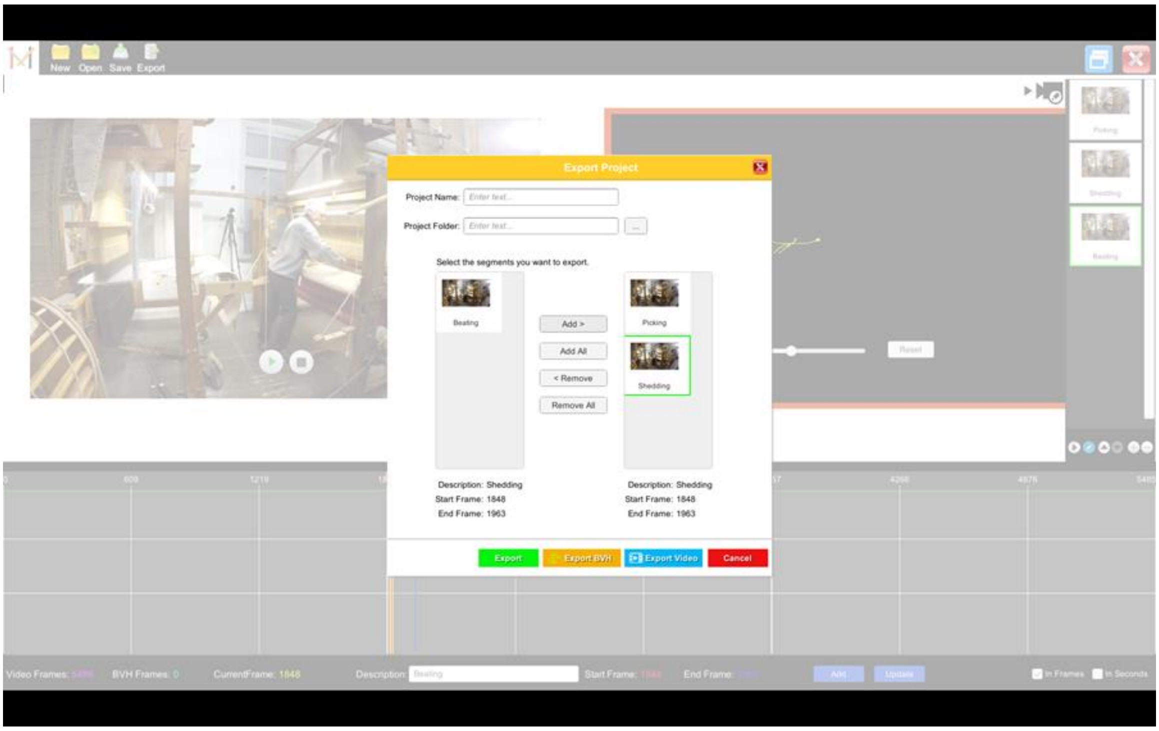
Video Frames: 5486 BVH Frames: 0 CurrentFrame: 2017 Description: Picking Start Frame: 2017 End Frame: 2131

☑ In Frames □ In Seconds

Figure 17. Select playback timing modality. (source: Mingei, 2019) [43]

5.1.3.15 Export AnimIO Project

To export an AnimIO project use the “Export” button at the top left side of the AnimIO. In the export window that appears, enter the project name, the target folder, and select the segments to be exported (see Figure 18). From the list that contains all the segments of the project, move the selected parts to the list with the segments to be exported. Use “Export” to export the project, “Export BVH” to export only the BVH file, or “Export Video” to export only the video file. To abort at any time use “cancel”.



Project Name: Enter text...

Project Folder: Enter text...

Select the segments you want to export.

Description: Shedding
Start Frame: 1848
End Frame: 1963

Description: Shedding
Start Frame: 1848
End Frame: 1963

Video Frames: 5486 BVH Frames: 0 CurrentFrame: 1848 Description: Beating Start Frame: End Frame:

☑ In Frames □ In Seconds

Figure 18. Export AnimIO Project. (source: Mingei, 2019) [43]

5.2 Implementation

AnimIO was developed in the Unity3D game engine and the code was written in C#. Two main components used is the ffmpeg library for video operations and FORTH's BVH Unity 3D package implemented as part of the Mingei project, used by various Mingei applications for BVH visualization tasks.

5.2.1 AnimIO UI implementation

AnimIO UI supports both 2D and 3D rendering. The 2D space is used for the general user interface implementation and the 3D space is used to display the BVH actor. **We opted using Unity3D[4] game engine for the implementation of AnimIO because it can easily combine those (2D and 3D) spaces together and it has a straightforward API for interaction between them. The resulting application is a stand-alone desktop application that hides from the user the implementation details of the Unity3D[5] game engine.**

However, in order to do the video manipulation and editing such as splitting a video to smaller parts or merging video parts together, we needed to use an external tool. **We ended up using FFMPEG[6] command line tool due to its efficiency and ease of use within Unity. Users are not required to run the FFMPEG executable for the command line. AnimIO automates this process by silently creating a FFMPEG processing thread that carries out the requested tasks without the user knowing that an external software is used.** No prior knowledge or expertise on FFMPEG is required by end users and its invocation does not affect the overall user experience offered by the software.

5.2.2 BVH Unity3D package

The BVH Unity 3D package is an essential part of the editor, this section explains the BVH file structure followed by Mingei and goes into the implementation of the BVH package for Unity3D.

The rationale for the implementation of the BVH Unity 3D package was related to the need to provide a reusable software component to various applications previewing or altering BVH animation files (e.g. the Motion Visualizer and Analyser presented in D5.5.). Through this package any application developed in Unity3D can integrate the functionality offered without the need for redevelopment of fundamental BVH operations.

[4] https://unity.com/

[5] https://unity.com/

[6] https://www.ffmpeg.org/

5.2.2.1 BVH Module implementation

The **BVH Module** is a Unity package, written in C#. The package contains several components that aid in the usage and modification of BVH files for motion capture animations.

The BVH Parser is used to load the BVH file and transform the data into data structures that can be used in Unity, for processing, modifying, visualizing the skeleton, and the motion data. Internally, a flat representation of the skeleton structure is created. Motion data are stored in depth-first per-channel animation curves. Each animation curve contains a specific joint's attribute values from the whole animation. For example, we have the animation curve for the Root joint's X-rotation channel, meaning that given a time value (e.g. 2 sec), we can evaluate what the rotation around the X axis was. Dividing motion data per-joint and per-channel in animation curves was a design decision based on Unity's animation system APIs. More on the animation systems used further below.

Considering that the BVH format does not include information about the scale and orientation of the joints (other than the forward direction of each bone), the parsing module accounts for different sources of BVH files (and therefore different conventions) using the **Convention Profiles**. Motion Capture in studios are set up in a way that scale is preserved. Motion Capture from video by the use of Computer Vision algorithms might not be able to infer the scale of things, thus the scale of the skeleton and the animations end up miniscule. These profiles contain a sequence of some supported transformations, needed for the data to make sense. In essence, these profiles allow conversion between Left/Right handed systems and/or definition of global or per-axis scaling factors. A direct result of this feature is that source files being provided can remain intact and no pre-processing step is needed per-file. A single convention profile per BVH Source is enough for all their subsequent BVH animations, provided the Source does not change its own conventions. BVH Sources can be a motion capture suit (and its provided software) or an image-based proprietary motion capture module.

Additionally, there is a **BVH Writer** for exporting either modifications to the animation data values or exporting time regions of the initial animation data, back to a BVH file.

Apart from the **Parser/Writer** components, there is also **Unity specific glue code available** that can generate and visualize the skeleton in a meaningful and intuitive way in Unity.

Motion capture data are denser than regular key-frame animations, so Unity's legacy animation system could not support them. Usage of Unity's new Animation APIs was thus selected, both for performance and flexibility reasons. Aside from compatibility with Unity's Timeline animation system, the resulting animation code is easily extendable to the point where new features like motion analysis or noise reduction is relatively easy to incorporate and have been, in projects where this module has been integrated.

Finally, the package offers a Demo scene, showcasing the components.

6. Conclusions & Future Work

The recordings for silk weaving, mastic cultivation, and glass-making were completed. Motion capture was to capture and archive the motions of the experts. The data were processed to remove noise and artefacts in order for the motion to be accurate and realistic. Additional measurements, such as the breathing volume of the assistant during the creation of the glass jug, were recorded to have a complete picture of the tasks. In general, the tasks were performed realistically given the time constraints. Indeed, certain steps of each process might need hours or are performed days apart. As a result, certain recordings do not have the duration of the actual task. This does not affect the resulting data, because the scope of the motion capture recordings was to exhibit accurate gestures. The data were delivered as the industry standard BVH files that contain the processed motion recordings, and XML files with the metadata of each trial.

For further post-processing of the MoCap files and video sources a BHV editor called Animation Studio was implemented. The main functionality in relation to Mingei regards: (a) preview of motion and video files, (b) synchronous preview, (c) segmentation of video or motion files, (d) synchronous segmentation and (e) annotation of segments. Using this editor, CH professionals can easily segment and annotate motion data and video streams so as to be linked with the CrO to formulate motion vocabularies. to represent schemas regarding how a craft is practiced.

Currently, all three tasks have been recorded for Mingei. Future work will focus on the creation of an annotated dataset in multiple file formats (bvh, video, depth, and xml). This dataset will contain the recordings in various stages, from raw recordings to animated avatars. Additional recordings may need to be done to enrich the data with additional repetitions. From those recordings, a workbench can also be made and released to the public. This will be a great outreach opportunity, but also allow those highly unique and dexterous motions to be used for different projects beyond the current scope of Mingei.

References

1. Brigante, C.M., Abbate, N., Basile, A., Faulisi, A.C., Sessa, S.: Towards miniaturization of a MEMS-based wearable motion capture system. IEEE Transactions on industrial electronics. 58, 3234–3241 (2011)
2. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Example of an AnimIO project file. [Screen Capture]. Retrieved through screen capture of a tcode editor, unpublished research results
3. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), AnimIO tool structure. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
4. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Create AnimIO project. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
5. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Load project. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
6. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Synchronized playback. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
7. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Adjust skeleton view. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
8. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Creating a segment (red-line: segment start, blue-line: segment end). [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
9. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Create Segment of the Video and BVH. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
10. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Update segment. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
11. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Playback segment. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
12. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Reorder segments. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
13. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Select playback timing modality. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
14. Chatziantoniou, A., Patsiouras, N., Partarakis, N., Zabulis, X., (2019), Export AnimIO Project. [Screen Capture]. Retrieved through screen capture of the prototype, unpublished research results
15. Lan, R., & Sun, H. (2015). Automated human motion segmentation via motion regularities. The Visual Computer, 31(1), 35-53.
16. García-Vega, S., Álvarez-Meza, A. M., & Castellanos-Domínguez, C. G. (2013, November). MoCap data segmentation and classification using kernel based multi-channel analysis. In Iberoamerican Congress on Pattern Recognition (pp. 495-502). Springer, Berlin, Heidelberg.
17. Schulz, S., & Woerner, A. (2010, July). Automatic motion segmentation for human motion synthesis. In International Conference on Articulated Motion and Deformable Objects (pp. 182-191). Springer, Berlin, Heidelberg.
18. Kadu, H., & Kuo, C. C. J. (2014). Automatic human mocap data classification. IEEE Transactions on Multimedia, 16(8), 2191-2202.17
19. Li, C., Kulkarni, P. R., & Prabhakaran, B. (2007). Segmentation and recognition of motion capture data stream by classification. *Multimedia tools and applications*, *35*(1), 55-70.Müller, M., & Röder, T. (2006, September). Motion templates for automatic classification and retrieval of motion capture data. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (pp. 137-146).
20. Olivas B., (2019), Figure 1. Breathing Recording of Glass Blowing. [Photo] Original unpublished work

21. Olivas B., (2019), Figure 2. Low pass filtering of the right forearm joint angles that indicate the flexion/extension Original unpublished results
22. Olivas B., (2019), Figure 3. Right forearm joint angles processed: ‘x’ corresponds to flexion/rotation, ‘y’ to supination/pronation, and ‘z’ to carrying angle. Original unpublished results
23. Olivas B., (2019), Figure 4.Posture correction with MotionBuilder: a) Erroneous posture, b) Posture corrected . Original unpublished work
24. Olivas B., (2019), Figure 5.Example of an XML file. Original unpublished work
25. Olivas B., Menychtas D., (2019),Table 1. Tasks before the jacquard weaving. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
26. Olivas B., Menychtas D., (2019), Table 2. Jacquard weaving with different loom sizes. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
27. Olivas B., Menychtas D., (2019), Table 3. Tasks related to the preparation of the tree before the mastic collection. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
28. Olivas B., Menychtas D., (2019), Table 4. Tasks related to the collection of the mastic. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
29. Olivas B., Menychtas D., (2019), Table 5. Three different techniques for the task of wounding the tree. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
30. Olivas B., Menychtas D., (2019), Table 6. Three different ways for the task of cleaning the mastic. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
31. Olivas B., Menychtas D., (2019), Table 7. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
32. Olivas B., Menychtas D., (2019), Table 8. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
33. Olivas B., Menychtas D., (2019), Table 9. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
34. Olivas B., Menychtas D., (2019), Table 10. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
35. Olivas B., Menychtas D., (2019), Table 11. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
36. Olivas B., Menychtas D., (2019), Table 12. Tasks recorded at CERFAV. [Screen Capture]. Retrieved through screen capture of the original, unpublished research results
37. Olivas B., Menychtas D., (2019), Table 13. Repetitions and mean durations of gestures extracted from the tasks recorded at HdS. Original, unpublished work.
38. Olivas B., Menychtas D., (2019), Table 14. Repetitions and mean durations of gestures extracted from the tasks recorded at Chios Mastic Museum. Original, unpublished work
39. Olivas B., Menychtas D., (2019), Table 15. Repetitions and mean durations of gestures extracted from the tasks recorded at CERFAV. Original, unpublished work.
40. Olivas B., Menychtas D. (2019), Table 16. Content description of the XML files. Original, Unpublished work.
41. Z. Cao, G. Hidalgo, T. Simon, S. E. Wei and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1302-1310, 2018.