

Deliverable 5.2 – Real time body tracking

Release Date	30.11.2019
Version	0.1
Dissemination Level	Public

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

Deliverable Number	D5.2
Deliverable Title	Real time human motion capture
Deliverable Type	Report
Dissemination Level	Public
Contractual Delivery Date	M12
Actual Delivery Date	29.11.2019
Work Package	WP5 – DEVELOP & EXECUTE: Build Toolkit and Platform
Authors	Sotiris Manitsaris, Alina Glushkova, Pablo Vicente Moñivar, Ioanna Thanou, Xenophon Zabulis, Nikolaos Partarakis, Evangelia Baka, Evropi Stefanidi
Number of pages (incl. cover)	99



Executive summary

This deliverable describes the work done in the task 5.2 for a) motion visualization of the expert gestures integrated in a 3D environment and b) real time body tracking, that consists on using RGB and depth images in order to estimate human body pose. In the framework of MINGEI project an interactive installation will be developed and installed in the museum. The user will be invited a) to observe expert's avatar performing the technical gestures, in a 3D environment and then b) to perform these gestures by him/herself by imitation and receive a feedback based on the quality of the performance (movement sonification). The data recorded under real conditions (described in the deliverable 5.1) will be used to visualise expert's motion and the work done permitting to achieve this visualisation is described in the first part of this deliverable. When inviting the user to live the movement sonification experience, it is necessary to track his body in real time, so that the system detects the articulations positions and compares them with expert's positions after the movement modelling process.

For the visualisation of expert motion Mingei proposes an approach where the practitioner is represented by a VH and objects through their 3D reconstructions. Craftspersons' actions are reproduced by animating the VH based on Motion Capture (MoCap) recordings, while the motion of machines and tools is induced by the human motion. The appropriate simulation of VHs is an important aspect, since crafts are practiced by humans and machines are designed for use by them. At the centre of the proposed approach is a conceptual, twofold decomposition of craft processes into actions, and of the machines used into components; this is essential in the systematic transfer of craft practice from the physical to the virtual domain, while retaining realism and allowing the semantic representation of craft processes. As an application of this approach, Mingei is implementing the MoViz platform, allowing users to create and experience craft usage scenarios. In particular, MoViz enables users to author their own scenes, where actions (from MoCap data) and machine parts (results of 3D reconstructions and 3D modelling) are assigned to VHs, in order to create Motion Vocabularies, representing the craft process. Users can also experience playback of the created scenes, through simulation in a VE, either in 3D or in Virtual Reality, where a training mode is also available. Through the aforementioned process, Mingei aims to deliver an efficient way of visualizing craft processes within VEs, thus increasing the usability and educational value of craft representation, and opening the way to a variety of new applications for craft presentation, education and thematic tourism, based on the value of tradition and intangible cultural heritage. Moreover, this work aims to contribute to the preservation and perpetuity of not only the craft, but also of the legacy of the practitioners, whose unique movements are preserved via the MoCap.

In order to achieve real time body tracking of the user, RGB-D images have been recorded for the three pilots during organised data acquisition sessions. The main goal of these recordings was to track human body and to extract features providing information about the precise positions of expert articulations in 2 or 3 dimensional space. Various challenges have been identified linked to the different nature of pilots, to the presence of multiple persons in the scene, to body occlusions etc. And possible solutions have been proposed such as recordings from different point of views, the possibility to extract less features etc. Their effectiveness has been evaluated through a comparison of gesture recognition results when using these different configurations. A comparative study is done when using 2D versus 3D coordinates and features

from only 2 joints (hands) instead of 7 (entire upper body). The real time body tracking results will be used to develop the interactive mechanism that will be integrated in the installation permitting to simulate expert gestures and receive a feedback on how well the gestures have been performed by the museum visitors.

Part A of this deliverable is structured as follows:

Section 1 provides an introduction to the methodology proposed by Mingei for the visualization of crafts in Virtual Environments (VEs), within which the practitioner is represented by a Virtual Human (VH) and objects through their 3D reconstructions. Practitioner actions are reproduced by animating the VH based on Motion Capture (MoCap) recordings.

Section 2 provides background work on 3D motion visualisation in general but also regarding relevant to the project approaches focusing on (a) authoring tools, (b) VR training and education, (c) Virtual Humans and (d) Human object interaction in virtual environments. Then the specific contributions of Mingei's approach is discussed in terms of the provision of (a) a comprehensive methodology for craft visualization, (b) an authoring platform for craft experiences, (c) methodologies for the visualisation of craft experiences in 3D and VR and (d) a method aiming to help in the presentation, representation and preservation of HCs.

Section 3 presents the technical details of the proposed methodology starting from craft understanding and resulting to the representation of the craft in 3D. In this steps how machines and affordances of tools are used is discussed, how MoCap data are translated to motion vocabularies and how this information is used for the association of craft actions with tools manipulation.

Section 4 presents the pilot studies Mingei decided to conduct during the first year of the project based on the data available from the pilot sites focusing on generic usage of handheld tools (which applies to all pilot sites) and the usage of machines selecting as a first use case the one of loom weaving.

Section 5 present the steps for the application of the proposed methodology in each pilot study.

Section 6 regards the implementation of an authoring, visualisation and training platform for craft experiences which incorporates parts of the proposed methodology and is used for the generation of the craft demonstrations.

Section 7 presents the effort done regarding the VR training approach followed by Mingei.

Section 7 presents the conclusion, challenges and open issues at the end of the first year of the project

Section 9 presents the planned work for the second year of the project in terms of improvement of methodology and infrastructure.

Part B of this deliverable is structured as follows:

Section 1 provides a short introduction to the topic of real -time body tracking and its role in the project.

Section 2 presents an overview of the existing methods for pose estimation, and the one used in MINGEI, based on a Deep Learning method is described. The gesture recognition methods permitting to validate feature's selection from pose estimation are also presented.

Section 3 refers to the sensors used to record images/videos from the 3 pilots, to the recording process and the datasets that have been created. These datasets contain RGB-D images that permit to extract from them information about precise positions of human body articulations. In this section, the challenges faced during these recordings are also discussed.

Section 4 provides a detailed description of the results and the comparative evaluation of human pose estimation done in 2 datasets/pilots. It also presents gesture recognition results while using different configurations (2D vs 3D, 2joints versus 7 joints) in order to identify the most meaningful features that are extracted from pose estimation.

Sections 5 presents the conclusions and the work to be done in futur focusing on the reuse of the datasets created.

This deliverable is submitted in the context of T5.2. 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 present the final Mingei solutions on Real Time Body Tracking integrated in the protocol and pilot experiences. This final version, will be submitted on M24.

Keywords

Avatar, 3D animation, pose estimation, depth map, gesture recognition, Hidden Markov Models, K-means

Document History

Date	Version	Author/Editor	Affiliation	Comment
21.08.2019	V0.1	Ioanna Thanou	ARMINES	Initial draft
08.09.2019	V0.2	Alina Glushkova	ARMINES	Second draft
09.10.2019	V0.3	Nikolaos Partarakis, Xenophon Zabulis, Evangelia Baka, Evropi Stefanidi	FORTH, MiraLab	Integration of Motion visualisation
01.11.2019	V0.4	Alina Glushkova	ARMINES	Third draft
25.11.2019	V0.5	Alina Glushkova	ARMINES	Final draft integration of review comments
27.11.2019	V0.6	Margherita Antona	FORTH	Final QA review
25-04-2020	R1	Ioanna Thanou Alina Glushkova	ARMINES	Updated draft
05-05-2020	R1.1	Evropi Stefanidi	FORTH	Updated draft
25-05-2020	R1.2	Nikolaos Partarakis, Xenophon Zabulis	FORTH	Final Draft

Abbreviations

FMC	Fundamental Machine Component
HC	Heritage Craft
MoCap	Motion Capture
MV	Motion Vocabulary
MVI	Motion Vocabulary Item
PSB	Project Steering Board
SEAG	Stakeholders Experts Advisory Group
VE	Virtual Environment
VH	Virtual Human
VR	Virtual Reality
ROS	Robot Operating System
RMPE	Regional Multi-Person Pose Estimation
R-CNN	Region Convolutional Neural Network

Table of Contents

Executive summary	2
Keywords	4
Document History	5
Abbreviations	5
1. Introduction	13
2. Background on 3D motion visualisation and Virtual Human Interaction	14
3. Virtual Humans Interacting with Handheld tools and Machines	21
4. Pilot Studies conducted during the first year of Mingei	28
5. Application of the proposed methodology to the two pilot studies	36
6. Implementing an Authoring, Visualization and Training Platform for Craft Experiences	48
7. VR training	58
8. Conclusions	59
9. Future Work	61
PART A - References	62
PART B – Detailed Table of Contents	69
1. Introduction	73
2. State of the art of pose estimation frameworks and gesture recognition methods	74
3. Data recording for Pose Estimation	82
4. Evaluation	89
5. Conclusions and Future Work	97
PART B - References	98

PART A – Detailed Table of Contents

1. Introduction	13
2. Background on 3D motion visualisation and Virtual Human Interaction	14
2.1. 3D Motion Visualization	14
2.2. Relevant Research Approaches	14
2.2.1. Authoring Tools for Experiences and Tool Usage Demonstration	14
2.2.2. Using Virtual Reality for Training and Education	15
2.2.3. Virtual Humans as Embodied Agents	16
2.2.4. Virtual Human-Object Interaction: Modeling and Affordances	17
2.3. Progress Beyond the State of the Art	18
2.3.1. Comprehensive Methodology for Craft Visualization in Virtual Environments	18
2.3.2. Authoring Platform for Craft Experiences	19
2.3.3. Visualization of Craft experiences in 3D and VR	19
2.3.4. Novel Method aiming to help in the Presentation, Representation and Preservation of Heritage Crafts	19
3. Virtual Humans Interacting with Handheld tools and Machines	21
3.1. Affordances	21
3.1.1. Handheld Tools Ergonomics	21
3.1.2. From Complex Machines to Simple Machines	23
3.2. Proposed Methodology	24
3.2.1. Craft Understanding & Conceptual Decomposition of the Craft	25
3.2.2. Identification of Craft Type	25
3.2.3. Decomposition of Machines	25
3.2.4. Motion Capture of the Practitioners	26
3.2.5. Association of Tools and Machine Parts with Motions	26

3.2.6.	Animating the Fundamental Machine Components	26
3.2.7.	Prerequisites and Mingei tool usage for Motion visualisation	26
4.	Pilot Studies conducted during the first year of Mingei	28
4.1.	Operating Handheld Tools	28
4.1.1.	Attachment of the Tool to the Virtual Human's Hand(s)	28
4.1.2.	Inducing the Tool's Motion	28
4.2.	Loom Weaving	29
4.2.1.	Pilot Objective	29
4.2.2.	Designing the Transition of Loom Weaving from the Physical to the Virtual World	29
4.2.3.	Loom Machine Parts	33
4.2.4.	Loom Weaving Materials & Products	34
4.2.5.	Association of Loom Machine Parts with Corresponding Motions	35
5.	Application of the proposed methodology to the two pilot studies	36
5.1.	Application of the Proposed Methodology for Handheld Tools: TooltY	36
5.1.1.	Step 1: Animation File for Human Motion	36
5.1.2.	Step 2: Tool Digitization	37
5.1.3.	Step 3: Editing of Animation Files in Animation Studio	37
5.1.4.	Step 4: Start TooltY	37
5.1.5.	Step 5: Selecting an Avatar for the Virtual Human	37
5.1.6.	Step 6: Application of Motion to the Virtual Human	38
5.1.7.	Step 7: Application of Motion to the Virtual Human	39
5.1.8.	Step 8: Addition and Spatial Registration of Room/3D Objects	41
5.1.9.	Step 9: Tool Manipulation from Human Motion	41
5.1.10.	Step 10: Play the Scene in 3D	42
5.2.	Application of the Proposed Methodology for the Loom Weaving Case	42

5.2.1.	Motion Capture	42
5.2.2.	Loom Model Acquisition	42
5.2.3.	Virtual Humans	42
5.2.4.	Motion Vocabulary	42
5.2.5.	Loom Machine Abstraction	43
5.2.6.	Association of Virtual Humans and FMCs	44
5.2.7.	Induced Machine Motion	46
6.	Implementing an Authoring, Visualization and Training Platform for Craft Experiences (MoViz)	48
6.1.	Requirements for the MoViz platform	48
6.2.	A Look at the Resulting Platform's UI	49
6.3.	Interacting with MoViz	55
6.3.1.	Overview	56
6.3.2.	Editing Mode	56
6.3.3.	Playback Mode	56
6.4.	Playback in 3D	57
7.	VR training	58
8.	Conclusions	59
8.1.	Challenges - Open Issues	60
9.	Future Work	61
PART A -	References	62

PART A - List of figures

Figure 1: Comparison of incorrect and correct postures for 3 handheld tools: a seesaw, pliers and a hammer.....	21
Figure 2: Power grip - thumb can be straightened as a precision component. (source: M. Patkin, “A Check-List for Handle Design,”) [71]	22
Figure 3: External precision grip. (source: M. Patkin, “A Check-List for Handle Design,”) [71]	22
Figure 4: Internal precision grip. (Source: M. Patkin, “A Check-List for Handle Design,”) [71].....	22
Figure 5: Handle diameter is correlated to the strength of the grip. (Source: M. Patkin, “A Check-List for Handle Design,”) [71]	23
Figure 6: Skilled control of fine movement - steadying two pinch grips together. (Source: M. Patkin, “A Check-List for Handle Design,”) [71]. The importance of studying these various aspects of handling and gripping handheld tools is therefore evident.	23
Figure 7: The 6 classical simple machines (source: Encyclopedia Britannica) [77]	24
Figure 8: The proposed methodology for Craft Visualization (source: Mingei, 2019) [89].....	25
Figure 9: The parts that a loom machine consists of. (source: LinkedIn, 2019) [90]	30
Figure 10: Co-design session at Haus der Seidenkultur (HdS), Krefeld, Germany. (source, Mingei, 2020) [91].....	30
Figure 11: Motion Capture sessions of a practitioner while loom weaving at HdS, Krefeld. (source: Mingei, 2020) [92].....	31
Figure 12: Basic loom components. (Source: Wikipedia, Compiled and edited by Mingei, 2019) [93]	31
Figure 13: Storyboard of the three stages of weaving and the machine parts involved. (source: Mingei, 2020) [94].....	32
Figure 14: Overview of the weaving process: steps, actions, and FMCs involved. (source: Mingei, 2020) [95].....	35
Figure 15: Overview of TooltY’s pipeline. (source: Mingei, 2020) [96]	36
Figure 16: Screenshot of the 2 Virtual Humans holding (a) a hammer and (b) scissors. (Source: Mingei, 2020) [97].....	38
Figure 17: Screenshots of a Virtual Human holding and operating a hammer. (Source: Mingei, 2020) [98]	39

Figure 18: Grip points, orientations and resulting attachment of hammer to the VH hand. (Source: Mingei, 2020) [99].....	41
Figure 19: Loom treadle model in its “max” and “min” positions, with joints visible. (Source: Mingei, 2020) [100].....	43
Figure 20: Loom beater model in its idle state and “max”, “min” positions, with joints visible. (source: Mingei, 2020) [101].....	44
Figure 21: Downloaded loom model vs model after editing (noticeable difference in the treadle mechanism) (source: Mingei, 2020) [102]	44
Figure 22: Visualization of the foot pressing the treadle. (source: Mingei, 2020) [103]	46
Figure 23: Attaching the hands on the beater. (source: Mingei, 2020) [104]	47
Figure 24: Attaching the hands on the shuttle. (source: Mingei, 2020) [105]	47
Figure 25: Visualization of the result: the VH is operating the loom. (source: Mingei, 2020) [106]...	48
Figure 26: Design 3.0 - MoViz Start Screen. (source: Mingei, 2020) [107].....	50
Figure 27: Design 3.0 - MoViz Start Screen - Add New Scene Selected. (Source: Mingei, 2020) [108]	50
Figure 28: Design 3.0 - MoViz Start Screen - Open Scene Selected. (source: Mingei, 2020) [109].....	51
Figure 29: Design 3.0 - MSE - Empty Scene. (source: Mingei, 2020) [110].....	51
Figure 30: Design 3.0 - MSE - empty Scene, hovering over Avatar in Library. (source: Mingei, 2020) [111]	52
Figure 31: Design 3.0 - MSE - Avatar Preview. (source: Mingei, 2020) [112].....	52
Figure 32: Design 3.0 - MSE - Avatar added to Scene. (source: Mingei, 2020) [113].....	53
Figure 33: Design 3.0 - MSE - Motion Vocabulary Item added. (source: Mingei, 2020) [114].....	53
Figure 34: Design 3.0 - MSE - Second Motion Vocabulary Item added. (source: Mingei, 2020) [115]	54
Figure 35: Design 3.0 - MSE - Motion Vocabulary Items are filled with Motions and FMCs, and Scene Objects have also been added. (source: Mingei, 2020) [116]	54
Figure 36: Design 3.0 - MSE - Preview of Animation Speed Selection Dropdown. (source: Mingei, 2020) [117].....	55

Figure 37: Design 3.0 - MSE - Preview of ability to delete an MVI by clicking on it. (Source: Mingei, 2020) [118].....55

Figure 38: Screenshots of the VR training module - showing to the user how to operate a hammer. (Source: Mingei, 2019) [119]. Online video at: <https://youtu.be/wpYxf-ZBFII>58

PART A - List of tables

Table 1: Decomposition of loom weaving into steps. (source: compiled by Mingei, 2019) [120]33

Table 2: Main machine parts involved in loom weaving (source: compiled by Mingei, 2019) [121]..34

Table 3: Materials and products identified for loom weaving. (source: compiled by Mingei, 2019) [122]35

1. Introduction

Driven by the main goal of Mingei to help in the preservation, dissemination and valorization of Heritage Crafts, this PART of D5.2 proposes a novel methodology for their visualization in Virtual Environments (VEs), within which the practitioner is represented by a Virtual Human (VH) and objects through their 3D reconstructions. **Practitioner actions are reproduced by animating the VH based on Motion Capture (MoCap) recordings.** The appropriate simulation of VHs is an important aspect, since crafts are practiced by humans and machines are designed for use by them. At the center of the proposed approach is a conceptual, twofold decomposition of craft processes into actions, and of machines into components that include their physical interface. This is essential in the systematic transfer of craft practice from the physical to the virtual world, while retaining realism. Additionally, this decomposition must be meaningful to allow the semantic representation of craft processes. Using this approach, a multitude of craft instances and machines can be modeled, by decomposing crafts to simple motion driven operations, and machines to Fundamental Machine Components.

In this context, an authoring and visualization tool has been developed to support the proposed methodology, called MoViz., which allows users to create and experience craft usage scenarios, and is comprised of two tools, integrated into one platform: (1) the Motion and Scene Editor (MSE) and (2) the Motion and Scene Player (MSP). The first is an authoring tool that enables users to create their own scenes, where actions and machine parts are assigned to Virtual Humans (in the form of 3D Avatars), so as to create Motion Vocabularies; in this way, users can recreate, reenact and represent available crafts in Virtual Environments. The second tool allows the playback of the scenes produced by MSE, either by simulation in a Virtual Environment (3D), or in VR, where a training mode is also available. The aforementioned process aims to deliver an efficient way of visualizing craft processes within VEs, targeted to increasing the usability and educational value of craft representation, and opening the way to a variety of new applications for craft presentation, education and thematic tourism, based on the value of tradition and intangible cultural heritage. In the first version of this deliverable, the focus is on the craft of loom weaving; however, the MoViz platform is generic, for representing any craft, after its decomposition according to our technique.

2. Background on 3D motion visualisation and Virtual Human Interaction

2.1 3D Motion Visualization

Regarding 3D Motion Visualization, various approaches have been used in different domains. “Key Probe” is a key frame extraction technique, relying on an appropriate algorithm for rigid-body and soft-body animations, which converts a skeleton-based motion or animated mesh to a keyframe-based representation [9]. Another methodology is “Action Snapshot”, which enables selecting a representative moment from a performance. This method is based on information theory, which automates the process of generating meaningful snapshots, by taking dynamic scenes as input, and producing a narrative image as output [10].

As another example, human motion visualization is used in the domain of sports to display 3D models of swimmers, by digitizing their motions and creating personalized virtual representations [11]. Furthermore, Lucent Vision is a visualization system developed for tennis, which uses real-time video analysis to extract motion trajectories and provides a variety of visualization options [12].

A prevalent technique in human motion visualization is also “action summarization”, as it can produce motion effects in still image frames. “Action Synopsis” takes human movements as input, encoded either as MoCap animations or videos, and presents motion in still images [13]. Another approach is the work in [14], which creates compact narratives from videos, by composing foreground and background scene regions into a single interactive image, using a series of spatiotemporal masks. Finally, depth information of animations assists summarization of 3D animations in a single image. In [15] a method is proposed, which extracts important frames from the animation sequence, based on the importance of each frame, depending on its contribution to the overall motion-gradient.

2.2 Relevant Research Approaches

In Mingei we argue that, to the best of our knowledge, no integrated environment exists for the representation and presentation of Heritage Craft processes and techniques via their authoring and visualization in 3D and VR Virtual Environments. Nevertheless, past research efforts have addressed relevant research topics, such as: (a) Authoring virtual experiences and visualizing the usage of tools, (b) Using VR for training, (c) Using VHS as Embodied Agents and (d) Virtual Human-Object Interaction. This section analyzes the aforementioned research topics, which provided inspiration for the work conducted while creating the MoViz research prototype.

2.2.1 Authoring Tools for Experiences and Tool Usage Demonstration

An authoring tool encapsulates various functionalities and features for the development of a specific product. They assist users in creating digital content, and encompass a wide variety of subjects, from eLearning authoring tools, to video capture and editing. One of their key features is that they enable users with little or no technical or programming expertise to utilize them. The software architecture of such a system empowers users with the necessary tools for content

creation to a) support users with intuitive and easy to use methodologies (visual scripting, editors), and b) provide advanced users with enhanced tools to extend the capabilities of the system.

Regarding already developed authoring tools for tool usage experiences, M.A.G.E.S [™] [16] is a platform facilitating just that. This platform proposes a novel VR SDK to deliver a psychomotor VR surgical training solution, supporting a visual scripting editor, scene customization plugins, custom VR software patterns and Unity editor tools for rapid prototyping of VR training scenarios. In addition, the M.A.G.E.S [™] platform offers ToolManager, a plugin designed for usage and manipulation of tools in VR environments. Utilizing ToolManager, a developer can transform any 3D model of a tool (pliers, hammer, scalpel, drills etc.) into a fully functional and interactive asset, ready to use in VR applications. After the tool generation, users can interact with it in the Virtual Environment and use it to complete specific tasks following recorded directions.

Another tool is ExProtoVAR [17], which allows the generation of interactive experiences in Augmented Reality for designers and non-programmers who do not have much technical background in AR interfaces. ARTIST [18] is a platform featuring methods and tools for real-time interaction between non-human and human characters to generate reusable, low-cost and optimized Mixed Reality experiences. This project proposes a code-free development environment for the deployment and implementation of MR applications, using semantic data from heterogeneous resources. Finally, RadEd [19] features a web-based teaching framework with an embedded smart editor to create case-based exercises for image interaction, such as attaching labels and selecting specific parts of the image and taking measurements.

2.2.2 Using Virtual Reality for Training and Education

Education can benefit from the use of technological tools, as they can make the learning process more enjoyable, effective, active and meaningful, as well as increase the interaction and motivation of learners [20] [21]. In particular, Virtual Reality has the potential to be successfully deployed as a training tool, as it has been shown by collaborative VR applications for learning [22], studies on the impact of VR in exposure treatment [23], as well as surveys on human social interaction [24]. In addition, VR offers the possibility to move safely around dangerous places, learning to cope with emotions while experimenting the best solutions remaining far away from the real dangers [25]. Moreover, trials have demonstrated that VR learning can also have a cognitive aspect [26] [27], which is important since cognitive learning is a type of learning that is active, constructive, and long-lasting [28].

Many existing VR platforms, such as Facebook Spaces, Bigscreen, VRChat, AltpaceVR, and Rec Room, provide a Virtual Environment to meet and discuss with other users while supporting collaborative mini-games for entertainment purposes; their main focus is on social interaction between the users, thus not addressing training and education. Furthermore, many VR simulators' primary goal is to provide training, neglecting the educational aspect [29]. There are however various applications focusing on education, such as a Virtual Art museum aimed at children [30], an immersive learning environment to teach the US army soldiers basic corrosion prevention and control knowledge [31], and a CAVE based system for teaching Mandarin [32].

Other research has focused on analysing the impact of training while immersed in a Virtual Environment. For instance, in [33] several VR-based experiments in training situations were

conducted with supply chain workers using different devices, while in [34] Architecture Engineering Construction specialists used VR for their professional training. In addition, [35] supports the design of architectural spatial experiences, and [36] describes an immersive (Avatar-based) 3D Virtual Environment for interpreting students with bilingual dialogues, in order to support situated learning in an institutional context.

2.2.3 Virtual Humans as Embodied Agents

Virtual characters, and in particular Virtual Humans, constitute an important aspect of 3D applications, due to our familiarity with human-like individuals. They are mainly used as narrators [37], virtual audiences [38], and in our case demonstrators of tool and machine usage, in the context of craft processes. We can distinguish two different styles in the representation of VHs: a) human-like and b) cartoon style Avatars, each one serving a different purpose and fitting into specific applications.

In the context of Virtual Environments VHs have already been utilized for explaining physical and procedural human tasks [39], simulating dangerous situations [40] and group and crowd behaviour [41], and assisting users during navigation, both by showing the location of objects/places, as well as by providing users with additional information [42]. Moreover, VHs offer a possible solution to the problem of unstructuredness [43], which requires the user to take the initiative both in exploring the environment and interacting with its parts. This lack of proper assistance clashes with the traditional learning scenario, where a real teacher structures the presentation of material and learning activities [44]. VHs can provide a solution by acting as an embodied teacher. While it is not feasible to provide a human tutor for every learner in the real world, embodied agents can aid anyone with access to a computer, enabling individualized instruction for a massive number of learners.

Virtual Humans are investigated in various research projects, with different systems offering conversational abilities, user training, adaptive behaviour and VH creation. For example, the ICT Virtual Human Toolkit [45] [46] offers a flexible framework for generating high fidelity embodied agents and integrating them in virtual environments. Embodied Conversational Agents as an alternative form of intelligent user interface are discussed in depth in [47], while in [48], Maxine is described, an animation engine that permits its users to author scenes and VHs, focusing on multimodal and emotional interaction.

Furthermore, Virtual Humans have been proven effective as museum storytellers, due to their inherent ability to simulate verbal as well as nonverbal communicative behaviour. This type of interface is made possible with the help of multimodal dialogue systems, which extend common speech dialogue systems with additional modalities similar to human-human interaction [74]. However, employing VHs as personal and believable dialogue partners in multimodal dialogs entails several challenges, because this not only requires a reliable and consistent motion and dialogue behaviour, but also appropriate nonverbal communication and affective behaviour. Over the last decade, there has been a considerable amount of progress in creating interactive, conversational, virtual agents, including Ada and Grace, a pair of virtual museum guides at the Boston Museum of Science [49], the INOTS and ELITE training systems at the Naval Station in Newport and Fort Benning [50], and the SimSensei system designed for healthcare support [51]. In the FearNot! application VHs have also been applied to facilitate bullying prevention education [52].

Regarding the different styles of VHs, users can more closely relate to human-like avatars, due to their natural appearance [53]. However, imperfect human-likeness can provoke dislike or strangely familiar feelings of eeriness and revulsion in observers, a phenomenon known as the uncanny valley [54]. The same principle is applied not only in the visual representation of a VH, but also in the way of moving in 3D space. Studies suggested that interaction and animation can overcome the valley in affinity due to matching and common human non-verbal cues [55]. VHs are usually more complex to animate, since they consist of humanoid skeletons with more joints than cartoon Avatars. The animation process needs to be precise and with high realism to avoid the uncanny valley effect. In contrast, cartoon style Avatars represent a simplified version of human characters, leading to a lower complexity in generation and animation processes. Cartoon Avatars are mainly used in applications designed with a similar cartoony style to match their environment. However, this design style reflects limitations on the usage of cartoon Avatars in realistic environments, making realistic-looking VHs ideal in such scenarios.

In Mingei we decided to use realistic, human-like Virtual Humans for a number of reasons. First of all, tools and machines are designed to be handled by humans, thus making the human hand ideal to interact with them in a natural way. Most cartoon style Avatars do not have five fingers, resulting in poor and unrealistic handling capabilities with tools. Additionally, we wanted to develop a realistic representation of tool and machine usage in the context of Heritage Craft processes, i.e. a simulation for interacting with tools and machines in Virtual Environments; thus, utilizing human-like Avatars was the preferred choice.

2.2.4 Virtual Human-Object Interaction: Modeling and Affordances

While there is a lot of research on the general topic of human-object interaction, such as for object recognition and detection [56], and action recognition [57], not a lot of it has focused on modelling the interaction between Virtual Humans and virtual objects. One of the most relevant works is the one conducted in [58], where they suggest including all the necessary information of how to interact with an object within its description, naming these kind of objects *smart*. To that end, a graphical user interface is employed, which allows the identification of object interaction features, e.g., moving parts and functionality instructions. Building on this idea of *smart objects*, [59] adds Artificial Intelligence Planning to the concept, in order to address adaptation to new situations and solving dynamic problems. Other approaches include object specific reasoning [60], according to which a relational table is created containing information about (i) the object's purpose and, (ii) for each object graspable site, the appropriate hand shape and grasp approach direction.

An important aspect to consider when modelling the interaction between Virtual Agents (Humans) and Objects is the object's affordances. According to [61], an affordance is an intrinsic property of an object. In the research field of Human-Computer Interaction (HCI) and Interaction Design, affordance is currently one of the most fundamental concepts [62]. The concept of affordances originates from ecological psychology, proposed by Gibson [63] [64] to denote action possibilities provided by the environment to the actor, and was introduced to HCI in the late 1980s by Norman [65]. In a broader sense, affordance is the functional classification of objects, which is a prevalent research topic in the domain of robotics and computer vision [66]. In order to detect an object's affordances, various approaches have been used, such as inferring them from human demonstration [67], or using attributes for fine (affordance related to core traits of an object, e.g., graspability, rollability) and high (e.g., drinkability or pourability of a glass) level affordance

detection [66]. As another example, [68] presents a weakly supervised approach for discovering all possible object functionalities, by representing each one by as specific type of human-object interaction, and evaluating image similarity in 3D in order to cluster human-object interactions more coherently.

In the context of Mingei, we do not infer the affordances of objects used in the crafts through Computer Vision because such approaches would not be the best road to follow, since HCs entail creativity, and special studying of the way that practitioners interact with each tool and machine part needs to take place. It is namely essential to involve the craftsman in the process of decomposing the craft to its essential actions and the machines used in their essential parts. We thus define the affordances of each Fundamental Machine Component, which also allows replicability of our approach.

2.3 Progress beyond the State of the Art

By reviewing the related work and state of the art, it appears that, until now, there does not exist a comprehensive approach that leads to the visualization of crafts in Virtual Environments, utilizing the practitioners' real movements. Various approaches have been developed for motion visualization, as it has been presented in the previous sections, but they are case-specific. We therefore aim to fill this gap, by proposing an integrated platform for the presentation of crafts in Virtual Environments. We argue that our methodology of decomposing (complex) machines into their functional components, inspired by the theory of simple machines, allows for generalization and facilitates replication. We also claim that the efficient visualization of Heritage Crafts can lead to their presentation in an attractive and engaging way to the general public, which can produce numerous benefits. The contributions provided Mingei in this context are discussed in more detail in the subsequent sections.

2.3.1 Comprehensive Methodology for Craft Visualization in Virtual Environments

In Mingei a novel methodology for the visualization of Motion in Virtual Environments has been formulated. Until now, interaction of Virtual Humans (VHs) with tools and machines has mostly been addressed through predefined animations of both the object and the VH, or through the usage of physics engines. **Our approach proposes the visualization of motion in the context of HCs, through Motion Capture of humans performing the craft, in combination with inducing the motion of the machine or tool that is used.** To that end, **we combine segmented MoCap animation files**, according to their conceptual decomposition, **together with articulations of the machines to their functional components**, and **create Motion Vocabularies**, representing the craft.

An integral part of the methodology is its branching depending on the use of (only) handheld tools or (also) of machines in its scope. In the latter case, we propose that the machines utilized in (Heritage) crafts can be decomposed to their functional parts, following the theory of simple machines [69], which in the context of Mingei we call Fundamental Machine Components. This allows understanding and generalization, as well as replication, since otherwise each unique machine instance would need to be atomically modelled. Each Fundamental Machine Component is then bound to a human motion, which provides the required data for the inference of its movement. Thus, both **kinetic properties and constraints of the machine are facilitated**, as well as

artificially generated constraints in the machine’s digital model, in the form of **boundaries that trigger collision detection mechanisms**, which allow to know when two (2) virtual objects are in contact.

2.3.2 Authoring Platform for Craft Experiences

To the best of our knowledge, no other platform exists that allows users to author their own scenes consisting of Virtual Humans and 3D-reconstructed craft objects (machines and tools), and enables the association of motions with the corresponding machine parts/tools, so as to recreate and re-enact scenarios of craft usage. Furthermore, the proposed platform allows even non-technical users to utilize the results of complex technological advancements, such as Motion Capture and 3D reconstruction, to author their own scenes. At the same time, it gives the opportunity to content owners, such as creative industries, museum curators and exhibitors without programming knowledge to utilize this tool to promote, share and disseminate their cultural content to the general public, aiming to provide engaging, entertaining and potentially educative content.

2.3.3 Visualization of Craft experiences in 3D and VR

Mingei proposes the visualization of the authored scenes in 3D, where the users can experience the process of the craft, re-enacted by the Virtual Human. Moreover, users can choose to view the created scene in VR, allowing for immersion and a closer view on the process, as well as choose to perform VR training. In the last case, the ideal motion trajectories of each tool are shown to the users, based on the motion that the craft practitioner performed; subsequently, the user’s trajectories are recorded while they are executing the movement, so as to compare them with the ideal ones, and provide them with feedback regarding their performance (accuracy, time, etc.). It should be noted that, regarding VR training, our contribution does not lie in the development of the underlying platform facilitating the creation of these VR training scenarios [70], but in its addition to MoViz and its use for training in the context of craft experiences, and in particular in those of Heritage Crafts.

2.3.4 Novel Method aiming to help in the Presentation, Representation and Preservation of Heritage Crafts

Through the development of the proposed methodology, as well as of the authoring and visualization platform for craft experiences, Mingei proposes a novel method for the presentation of Heritage Crafts, aiming to aid in their representation and preservation, from which multiple user groups can benefit: (i) craftspersons whose work will be preserved and represented, (ii) local communities in which the craft is practiced, (iii) museum curators and exhibitors for the presentation of various traditional crafts and (iv) people who do not necessarily possess knowledge or specialization regarding HCs, who are however interested in a HC and wish to learn more about it (e.g., tourists, teachers, school groups, craft enthusiasts).

Using the proposed approach, it is possible to model a multitude of craft instances and machines, by decomposing crafts to simple motion driven operations, and machines to fundamental machine components (FMCs). We assume that each craft process can be modelled as a series of actions, which we call Motion Vocabulary Items (MVIs); thus, the combination of the MVIs forms the Motion Vocabulary, which represents the craft as a series of actions. Through this process we aim

to deliver a more efficient way of visualizing craft processes within Virtual Environments, increasing the usability and educational value of craft representation, and thus opening the way to a variety of new applications for craft presentation, education and thematic tourism, based on the value of tradition and intangible cultural heritage.

3. Virtual Humans Interacting with Handheld tools and Machines

In this chapter the Mingei proposed process for Virtual Humans Interacting with Handheld tools and Machines is presented. In this process the **starting point is always the 3D information stemming from the MoCap of the human practitioner** and the **ending point is the re-enactment of the craft in a Virtual Environment where Virtual Humans Interacting with Handheld tools and Machines**.

3.1 Affordances

In the context of Mingei, a methodology is proposed for the visualization of Virtual Humans while performing craft processes. This essentially entails the visualization of their interaction with the handheld tools and machines that are part of the craft processes. The following sections present how we model the objects used in two (2) different cases of craft types, that is (i) crafts that use (only) handheld tools and (ii) crafts that (also) use machines.

3.1.1 Handheld Tools Ergonomics

When discussing the operation of a handheld tool, one important factor is to know how to hold it, or “grip” it. In particular, and since we aim to present a craft as accurately as possible, especially important for educational and training purposes are the ergonomics of how to hold and use a tool. Namely, we need to examine how humans grip their handles. As an example, Figure 1 below shows correct and incorrect grip postures for three (3) different tools. According to [71], there are various types of hand grips, and different classifications of them exist. One approach is to classify them in the following six (6) types: (a) Power grip, (b) Pinch, (c) External precision grip, (d) Internal precision grip, (e) Ulnar storage grip and (f) Other Power grip. Some indicative images are depicted in Figures 2-5 [71].

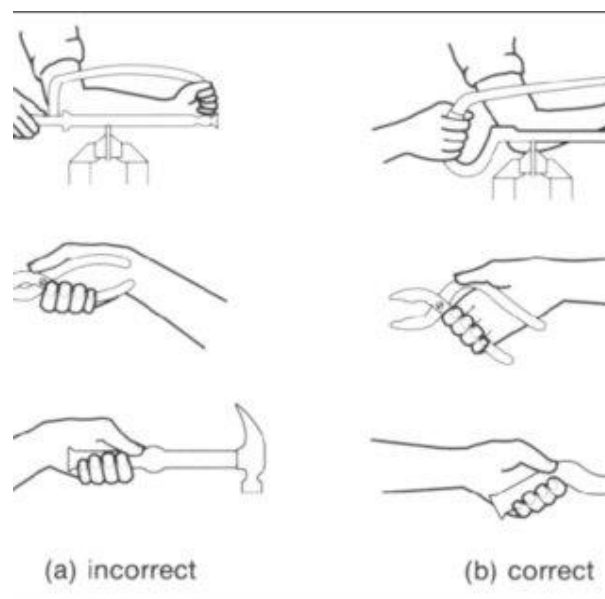


Figure 1: Comparison of incorrect and correct postures for 3 handheld tools: a seesaw, pliers and a hammer.

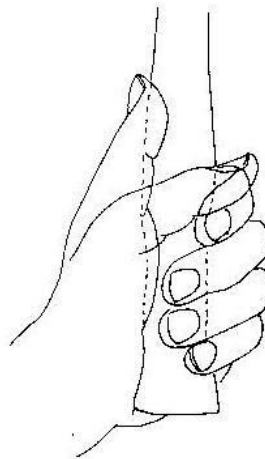


Figure 2: Power grip - thumb can be straightened as a precision component. (source: M. Patkin, "A Check-List for Handle Design,") [71]

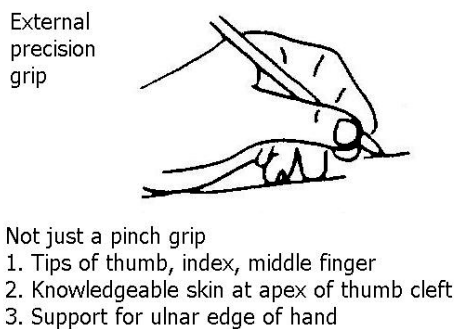


Figure 3: External precision grip. (source: M. Patkin, "A Check-List for Handle Design,") [71]

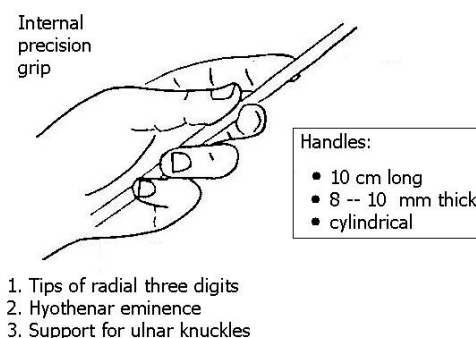


Figure 4: Internal precision grip. (Source: M. Patkin, "A Check-List for Handle Design,") [71]

Therefore, it becomes apparent that there are some guidelines that need to be followed regarding handle-design, that also apply to their presentation, so that the usage of tools is demonstrated in a correct and accurate manner. According to [71], some of these guidelines include size, which can for instance affect the strength of the grip (Figure 5), as well as the skill of the person who will use the tool. An example of the latter is steadying the two hands together to thread a needle, a simple element of movement which most skilled sewers are quite unaware of (Figure 6).

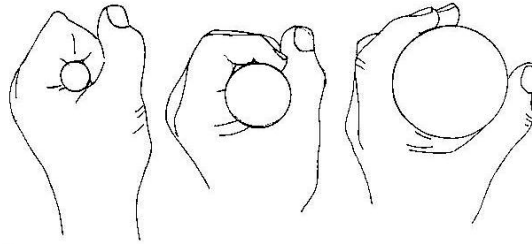


Figure 5: Handle diameter is correlated to the strength of the grip. (Source: M. Patkin, "A Check-List for Handle Design,") [71]

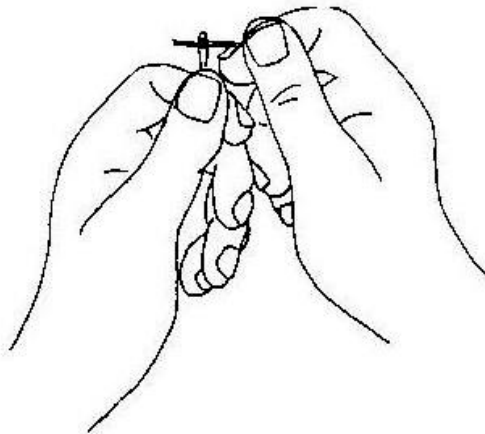
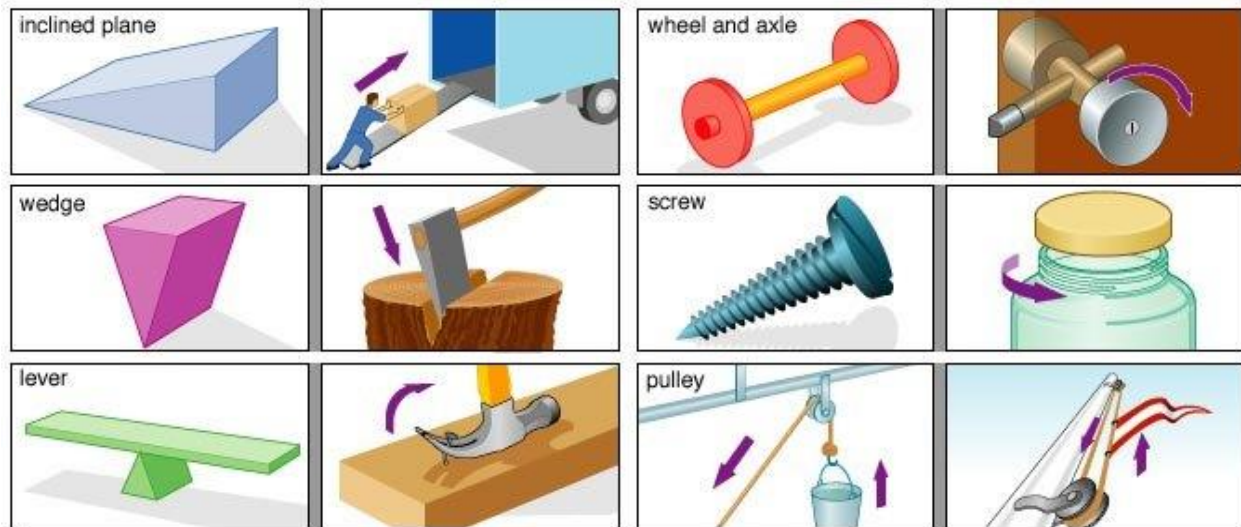


Figure 6: Skilled control of fine movement - steadying two pinch grips together. (Source: M. Patkin, "A Check-List for Handle Design,") [71]. The importance of studying these various aspects of handling and gripping handheld tools is therefore evident.

3.1.2 From Complex Machines to Simple Machines

In order to model the machines used in each craft, we propose their decomposition into their functional parts, inspired by the theory of Simple Machines. A simple machine is a mechanical device that changes the direction or magnitude of a force [72]. Simple machines can be regarded as the elementary "building blocks" of which all more complicated machines (sometimes called "compound machines" [73]) are composed [74] [75]. Usually the term refers to the six classical simple machines that were defined by Renaissance scientists [76], also depicted below in Figure 7:



© 2006 Encyclopædia Britannica, Inc.

Figure 7: The 6 classical simple machines (source: Encyclopedia Britannica) [77]

- **Inclined Plane:** A simple machine that has a gently sloped surface so it can be used to move objects upwards with less force (e.g., a ramp).
- **Wedge:** A simple machine that gets thinner at one end that is used to split material such as wood (e.g., a knife).
- **Lever:** A plank that rests on something underneath and moves up and down (e.g., a seesaw).
- **Wheel and axle:** A wheel and axle is made up of a circular frame -the wheel- that revolves on a shaft or rod -the axle (e.g., car tires).
- **Screw:** An inclined plane wrapped around a center rod (e.g., a spiral staircase).
- **Pulley:** A wheel and rope that can change the direction of a force (e.g., a flagpole uses a pulley to raise the flag).

In the context of this work, a relevant issue is to find a generic way to decompose complex machines, so that we do not have to model each machine, but rather have a generic decomposition modeling. Thus, as for example the basic mechanism of a bicycle consists of wheels, levers, and pulleys [78] [79], propose the decomposition of the machines involved in Heritage Crafts into their functional parts. We call these functional parts of the machines Fundamental Machine Components (FMCs). For instance, for the craft of weaving with a loom machine, the FMCs are the (i) treadle, (ii) shuttle and (iii) beater, where the treadle (pedal) of the loom can be decomposed to the following simple machines: (i) a lever with (ii) a pulley.

3.2 Proposed Methodology

This Section describes the general proposed methodology. Given a craft instance, which we want to transfer from the “real” to the “virtual” world, we propose the following pipeline, visualized below in Figure 8:

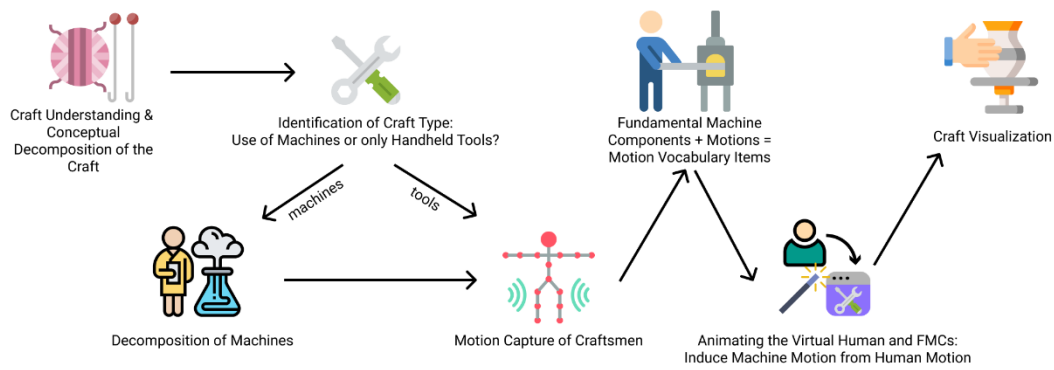


Figure 8: The proposed methodology for Craft Visualization (source: Mingei, 2019) [89]

The following sub-section provides a step by step presentation of the proposed pipeline.

3.2.1 Craft Understanding & Conceptual Decomposition of the Craft

First of all, it is necessary to study and understand the craft instance in question. In this respect, it is essential that craftspersons are centrally involved (e.g., by interviewing them, holding co-creation collaborative sessions etc.), so as to provide functional insight and emic understanding of the represented process (co-creation sessions and outcomes are reported in D1.1). Subsequently, the craft can be conceptually decomposed into its essential composite actions, thus presenting an abstraction of the movements comprising the craft (MoCap process and decomposition reported in D5.1). We call each composite action a Motion Vocabulary Item (MVI), so that all the MVIs together form the Motion Vocabulary of a craft, which can represent the craft as a whole.

3.2.2 Identification of Craft Type

The next step is to categorize the craft instance in question: are machines used during the craft execution, or do this craft only include the utilization of handheld tools? This is an important classification to be made, since if the craft involves machines, they need to be decomposed in their functional parts, inspired by the theory of Simple Machines. In both cases, it is necessary to identify the correct grip postures, point(s) and orientation of the handheld tools, as well as the correct usage and attachment to the body part that should operate each functional part of the machine in general. This means that for both handheld tools and for machine parts, it is essential to identify their affordances, i.e. their properties that show the possible actions that can be performed with them, suggesting how users may interact with them.

3.2.3 Decomposition of Machines

In case that the craft in question does not (only) involve handheld tools, but (also) machines, they need to be decomposed into Fundamental Machine Components. We achieve that by segmenting them into their functional parts, thus providing an abstraction of the machine. Especially in the case of complex machines, this approach presents a generic solution, since otherwise each machine would comprise a unique case, which would need its own modeling. We call these functional parts of the machines Fundamental Machine Components (FMCs). In the case of handheld tools, since they already constitute a “single unit”, the actual tool is also an FMC in the context of the proposed methodology.

3.2.4 Motion Capture of the Practitioners

After studying and analyzing the craft, and its conceptual decomposition, the movements of the practitioners while they are performing it need to be captured. This is achieved by performing Motion Capture sessions, the result of which are MoCap files, representing the human movement (presented in D5.1).

The value of this decomposition is twofold: first of all, it allows for the **exact and correct movements of the practitioners to be digitized, so that they can subsequently be used for animating the Virtual Humans that will reenact the craft**; secondly, **the practitioner's motions during the execution of the craft are digitally recorded thus contributing to the craft's preservation**. It is important to note that, since Motion Vocabularies can be used to create sequences that encode different actions and procedures, they can encode a wider variation of actions and combinations of actions than the initial MoCap data used for their implementation.

3.2.5 Association of Tools and Machine Parts with Motions

After having at hand the **decompositions of both the actions and the machines used (FMCs)**, we **associate the actions with the corresponding machine parts or tools used during their execution**. Thus, the MV is formed, where each MVI essentially consists of a motion, bound with its corresponding FMC.

3.2.6 Animating the Fundamental Machine Components

The Virtual Humans' movements to reenact a craft in the **Virtual Environment** are animated based on the data of the MoCap files. However, **we do not have MoCap for the FMCs (tools or machine parts); instead, we induce the machine motion from the human motion**. Thus, having the human MoCap, a digitized version of the FMC (e.g., via 3D reconstruction), and the conceptual decomposition of the craft into its essential actions and FMCs, we can infer the FMCs' movements and visualize the whole Motion Vocabulary that these ingredients compose. We argue that **this approach allows for generalization of the process, and bypasses potential problems that would occur by trying to perform Motion Capture on tools and machines**. Such problems include obstruction of using the machine or tool because of the presence of sensors, which might make it unusable or alter the way that the practitioner is holding, handling and operating it.

3.2.7 Prerequisites and Mingei tool usage for Motion visualization

This sub-section summarises the prerequisite implied by the aforementioned process

3.2.7.1 Motion Vocabularies

A Motion Vocabulary Item (MVI) is used in the context of this research work to represent an instance of a movement that is encoded in a BVH file and can be used to represent a specific action or part of an action. MVIs can be combined and interleaved to represent entire procedures and are considered building blocks of a Motion Vocabulary (MV). The MV in turn can be used to create "sentences" that encode different actions and procedures. As a result, the MV can be used to

encode a wider variation of actions and combinations of actions than the initial MoCap data used for its implementation.

For editing the MoCap animation files, we use a BHV editor, developed in-house, called Animation Studio¹ (reported in D5.5). Animation Studio 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, but does not stream for video and motion capture. This tool can be used for motion segmentation to create MVIs. The entire process is presented in D5.1.

3.2.7.2 Motion Capture

There are various ways to procure a MoCap, and all of them can work in our context. **For the purposes of the first version of this deliverable , we used MoCap files that were the product of motion capturing with NANSENSE®² R2 motion capture suit in the context of motion capture sessions (reported in D5.1).** Furthermore, the solutions presented in this chapter can employ outputs from the hands and body visual tracking methods implemented in T5.4 and reported in D5.4.

3.2.7.3 Implementation of avatars

The Avatar representing the VH plays a very important role in our concept, as it is the actor executing the movements representing the craft, thus bringing the whole process to life. For Mingei **we are facilitating the Unity3D game engine, and thus Avatars created using a plethora of 3D Computer Graphics and Animation Creation editors can be imported (e.g. 3DStudioMax, Fusion 360, etc.).** For the purposes of this research work, Poser Pro 11³ as well as Adobe Fuse [3] were employed for the creation of VHs.

3.2.7.4 Digitization of tools and machines

Another prerequisite is the digitized form of the tool(s) that will be used. For this, various approaches for 3D reconstruction for digitization can be used (a review can be found at D1.3). The result of the digitization is a 3D model that represents the surface geometry and appearance of the object and is, typically, encoded as a textured mesh of triangles (a summary of Mingei Digitisations is presented in D2.2.).

¹Presented in the public deliverable D1.3 “Scientific protocol for craft representation” of Mingei (European Union's H2020 research and innovation program under grant agreement No. 822336) to be published at Mingei’s website www.mingei-project.eu after approval by EC

² <https://www.nansense.com/>

³ www.posersoftware.com

4. Pilot Studies conducted during the first year of Mingei

Based on the proposed categorization of crafts depending on the usage of (only) handheld tools or (also) machines, we present one pilot for each of these cases: (i) the case of the handicraft of operating a hammer (similar for all simple hand-held tools), and (ii) the pilot case of the heritage craft of loom weaving.

For the first case, at the time of definition of the prototype no animation data from simple hand-held tools were available as the Mastic Pilot recording would happen in September 2019. To this end based on craft understanding process of mastic and the collection of knowledge regarding tools used during cultivation it was decided that a simple hand held tool, held by a grip on its side and moved with the entire arm is a good starting point for simulating the usage of such tools. Later on in the course of the project this development will also take into account MoCap data from the mastic pilot thus aligning the developments with the project objectives. The proposed methodology is generic so no great adjustments are foreseen.

The following two (2) sections describe these cases.

4.1 Operating Handheld Tools

The first pilot regards operating simple handheld tools, such as a hammer. In this case, which falls under the category of handheld tools, there is no need for a decomposition of the tool used - the tool itself is a Fundamental Machine Component.

4.1.1 Attachment of the Tool to the Virtual Human's Hand(s)

Nevertheless, as described in the relevant sections of the proposed Methodology (Chapter 3), the craft needs to be studied and comprehended, with special focus on the correct grip posture of the hand, so to achieve its correct attachment to the hand of the Virtual Human in the VE. In more detail, the definition of the following is necessary:

- A grip point \mathbf{p}_a on the hand of the Virtual Human that will operate the tool.
- A preferred grip on the tool, denoted as \mathbf{g} .
- Two (2) points on the front and back faces of the desired grip point of the tool, denoted as \mathbf{p}_f and \mathbf{p}_b , respectively.
- A grip center \mathbf{p}_c , which is automatically calculated as the mid-point of \mathbf{p}_f and \mathbf{p}_b .
- The projection of \mathbf{p}_c to the top side of the bounding box of the tool \mathbf{o} , denoted as \mathbf{p}_g , also automatically calculated. In this context, the bounding box of the tool can be defined as follows: for a point set in the 3D VE, we define it as the box with the smallest measure within which all the points of the tool lie. In simpler words, it is a virtual box surrounding the tool.

4.1.2 Inducing the Tool's Motion

Apart from the correct attachment of the handheld tool to the VH's hand(s), the second important task is inducing the tool's movement from the human motion. The inference of the animation of the tool also depends on the tool itself, i.e., if the tool is deformable or not. In more detail, there is

a methodological difference in our approach if the tool at hand is for instance in the category of (i) pliers, or (ii) a hammer.

Namely, in the first case, it is true that the movement of the tool will be directly induced as a result of the motion of both hands: since each hand holds a side of the tool's handle, as the hands open and close, the handles as well as the blades of the tool move accordingly. In the second case, however, since the hammer is a non-deformable tool, it will simply follow the motion of the hand that is operating it. For this case it simply suffices to have a correct grip of the handle of the hammer, and then apply the appropriate transformations to the tool as an entire object, so that its movement follows that of the hand.

However, in the case of deformable objects, additional configuration is required. In particular, joints need to be added to each deformable part of the tool, and the inference of the animation is a more complex process. Namely, the appropriate transformations need to be applied to the tool as a whole, so that its movement follows that of the hand, but also a second layer of animation needs to be added, so that the tool deforms according to the motion of the fingers. To that end, additional animation/Motion Capture files are needed, that will concern the motion of the fingers.

In the context of Mingei, and **during the first year of the project, we have actually implemented the complete methodology on only the first category of tools, i.e., the non-deformable ones**. This was due to the fact that for the moment we do not possess accurate animation files for the movement of the fingers in these tasks, while on the other hand we will present our work on deformable machine parts instead of tools in the context of the loom weaving pilot. Nevertheless, we also experimented with the operation of scissors, which belong in the second category, by the right hand of a VH, as an initial step towards exploring deformable handheld tools. To that end, we also created the corresponding animation of the tool, even if the grip on the tool was not entirely correct, since the appropriate animation was missing.

4.2 Loom Weaving

4.2.1 Pilot Objective

In the context of the loom pilot, our purpose is to model the act of loom weaving in a Virtual Environment, demonstrating it through a Virtual Human, so that users are able to experience and learn about its fundamental steps and process. The craft of loom weaving is essentially comprised of 3 basic motions (MVIs): shedding, picking, and battening. Thus, according to the proposed decomposition method, each one of these actions constitutes a Motion Vocabulary Item that, together, form the Motion Vocabulary of loom weaving. Depicted in Figure 9 are all the parts of a loom:

4.2.2 Designing the Transition of Loom Weaving from the Physical to the Virtual World

Regarding the conceptual decomposition of the craft, it is essential that craftspersons are centrally involved in it, so as to provide functional insight and emic understanding of the represented process. Within the context of the Mingei, we collaborated with the practitioner community of the

Association of Friends of Haus der Seidenkultur (HdS), Krefeld, Germany⁴ (Figure 8). HdS provided descriptions and testimonies, and allowed us to record functional demonstrations (MoCap, Video) of the practitioners, in order to perform careful observation and analysis of the craft, as well as acquire the necessary Motion Capture files for reenacting the craftspeople's movements (Figure 11). At the same time, collaborative sessions enabled craft understanding and provided insight from the perspective of the practitioner, towards a meaningful decomposition. Context definitions were then created, which are provided below and are depicted in Figure 12.

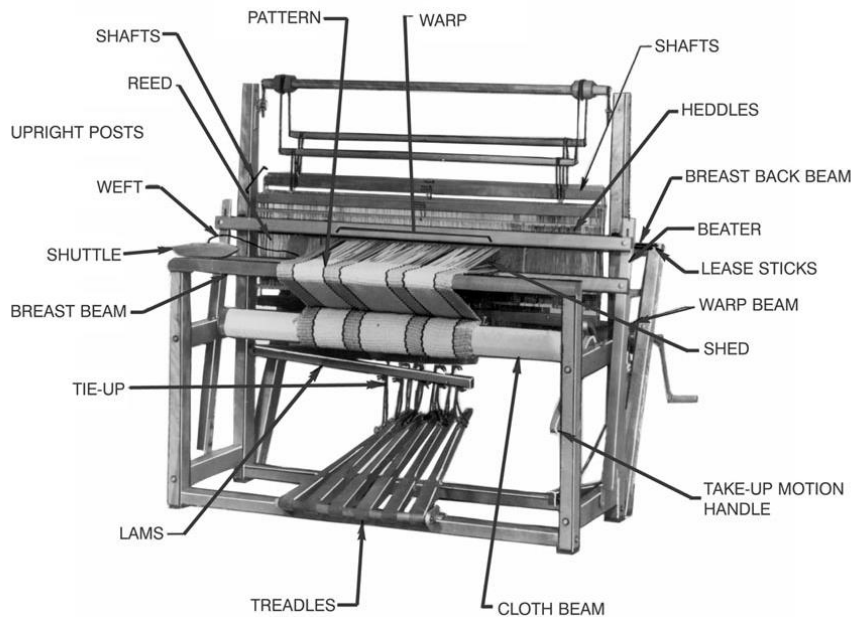


Figure 9: The parts that a loom machine consists of. (source: LinkedIn, 2019) [90]



Figure 10: Co-design session at Haus der Seidenkultur (HdS), Krefeld, Germany. (source, Mingei, 2020) [91]

⁴ <https://seidenkultur.de/>

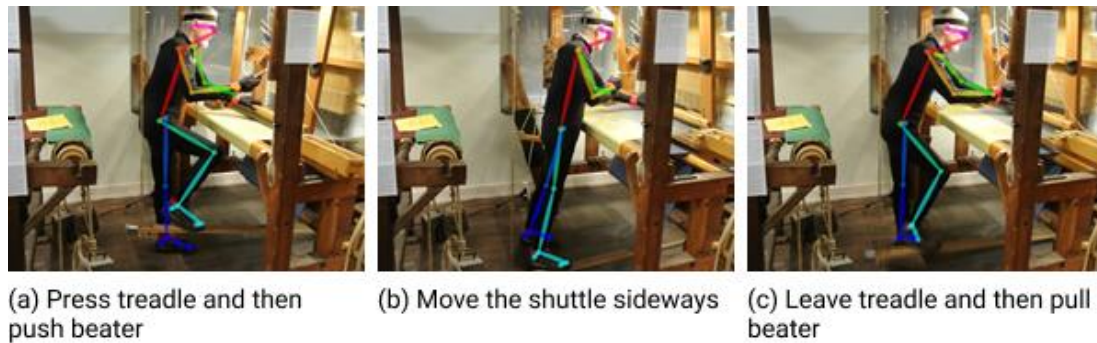


Figure 11: Motion Capture sessions of a practitioner while loom weaving at HdS, Krefeld. (source: Mingei, 2020) [92]



Figure 12: Basic loom components. (Source: Wikipedia, Compiled and edited by Mingei, 2019) [93]

Definitions:

- **Yarn** is a continuous length of interlocked fibers, produced by spinning fibers into long strands.
- **Warp and Weft** are the horizontal and vertical threads of a fabric.
- **Weaving** is the process of yarn transformation to fabric; vertical warp threads (warps) are held in tension on a loom, while weft is perpendicularly interlaced, fastened in-between elevated (upper) and lowered (lower) warps. The configuration of upper and lower warps (or the weave of the fabric), determines the structure of the woven fabric.
- **Shed** is the space due to the temporary separation of upper and lower warps.
- A **treadle** is a loom lever that mechanizes shed creation.
- A **shuttle** is a device used to interlace weft through upper and lower warps.
- Finally, a **beater** is a tool used to fasten the weft to the warp. Each thread of weft is fastened by a beat of the beater [80].
- A **loom** is a piece of machinery that facilitates weaving: it retains warps at tension, to facilitate the thread-by-thread interlacement of weft through them. There are several types of looms. In the conventional loom, weft is introduced using a shuttle.

In the case study, the weaving process was decomposed into 3 actions, repeated for each thread of weft [80]: (i) **Shedding**: warp threads are separated to form a shed, (ii) **Picking**: weft is passed across the shed using the shuttle, and (ii) **Beating**: weft is pushed against the fabric using the beater. Thus, the decomposed loom interface components are the shuttle, treadle and beater (Figure 11). Initially, textual descriptions were created collaboratively for each action, which also identified the machine interface components and human body parts used to operate them. We thus developed an analytical way to visually and textually represent a process comprised of actions that are performed on objects and machine interface components. In this collaborative process, the need for a representation that is intuitive to the practitioner and analytical enough for a semantic representation of the process was identified. To that end, storyboards [81] were selected as a methodological approach to address this need. The loom weaving process was encoded as a sequence of actions and reviewed by the community of practitioners, finally producing the storyboard visible below in Figure 13.

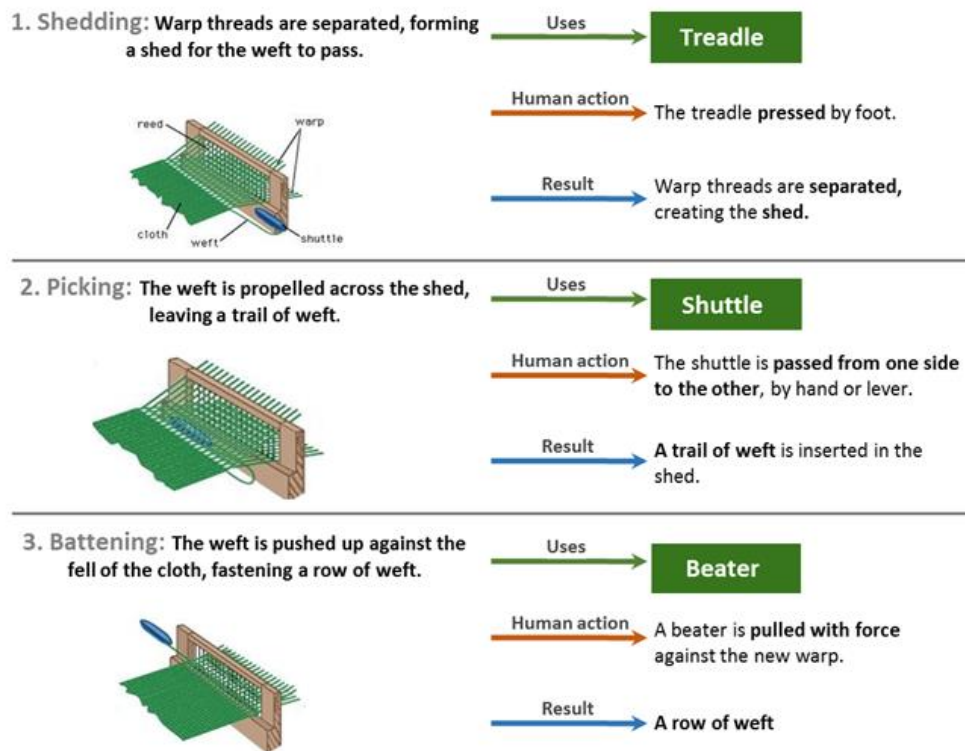


Figure 13: Storyboard of the three stages of weaving and the machine parts involved. (source: Mingei, 2020) [94]

This decomposition of the weaving process contains the interplay between human motion and components of the physical interface of the machine. To meaningfully represent the machine interface, it was decomposed in elementary components, according to our methodology (Fundamental Machine Components). Table 1 below shows the decomposition of loom weaving and its parts. In the figures, dashed lines plot the feasible induced motion trajectories.

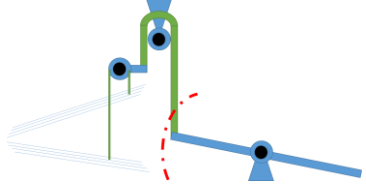
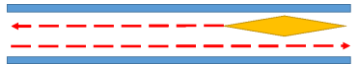
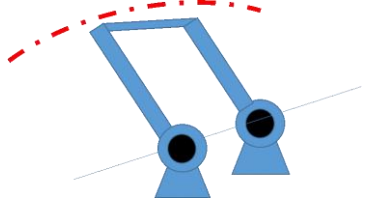



Steps	Action	Result	Design of the FMC ^a
Shedding	Treadle is pressed by foot .	Warp threads are separated by the press of the treadle .	
Picking	The shuttle is passed from one side to the other by hand .	A row of weft is created by a pass of the shuttle .	
Battening	The beater is dragged with force on the new warp.	Weft row completed using the beater .	

Table 1: Decomposition of loom weaving into steps. (source: compiled by Mingei, 2019) [120]

4.2.3 Loom Machine Parts

The machine parts relevant to the primary motions of the loom we identified are presented in Table 2 below:

Treadle: a loom lever that mechanizes shed creation.	
Shuttle: a device used to interlace weft through upper and lower warps.	
Reed: a metallic comb which is fixed to the sley with a reed cap.	



<p>Beater: a tool used to fasten the weft to the warp.</p>	
<p>Cloth beam/roller: the cylinder on which the resulting cloth is wrapped and collected.</p>	

Table 2: Main machine parts involved in loom weaving (source: compiled by Mingei, 2019) [121]

From these, three (3) constitute the Fundamental Machine Components (i.e., the treadle, shuttle and beater), since they are the ones that are operated by humans. The reed and cloth roller may constitute basic parts of the machine, but they are not directly manipulated by the craftspersons.

4.2.4 Loom Weaving Materials & Products

The materials and products associated with loom weaving are visible in Table 3 below:

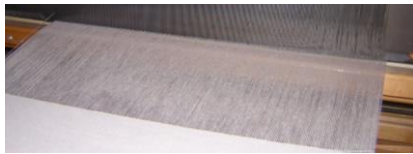
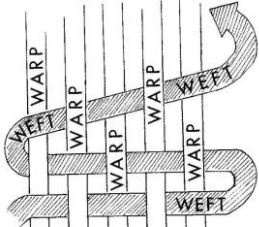
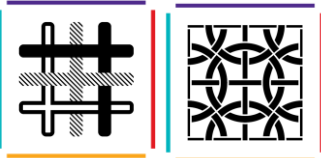
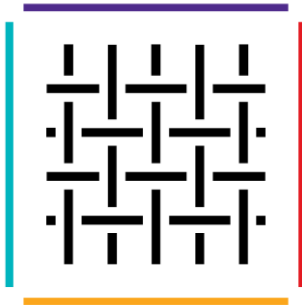
<p>Warp threads (representation for each state: raised, lowered)</p>	
<p>Warp: the threads on a loom over and under which other threads (the weft) are passed to make cloth</p>	
<p>Weaving product: Cloth</p> <p><i>Textile can also be colored and patterned.</i></p> <div data-bbox="153 1787 475 1944">  </div>	

Table 3: Materials and products identified for loom weaving. (source: compiled by Mingei, 2019) [122]

4.2.5 Association of Loom Machine Parts with Corresponding Motions

After having presented all the machine parts, materials, products, and actions associated with loom weaving, we arrive at the following categorization and association of fundamental machine components (FMCs) and motions (Figure 14):

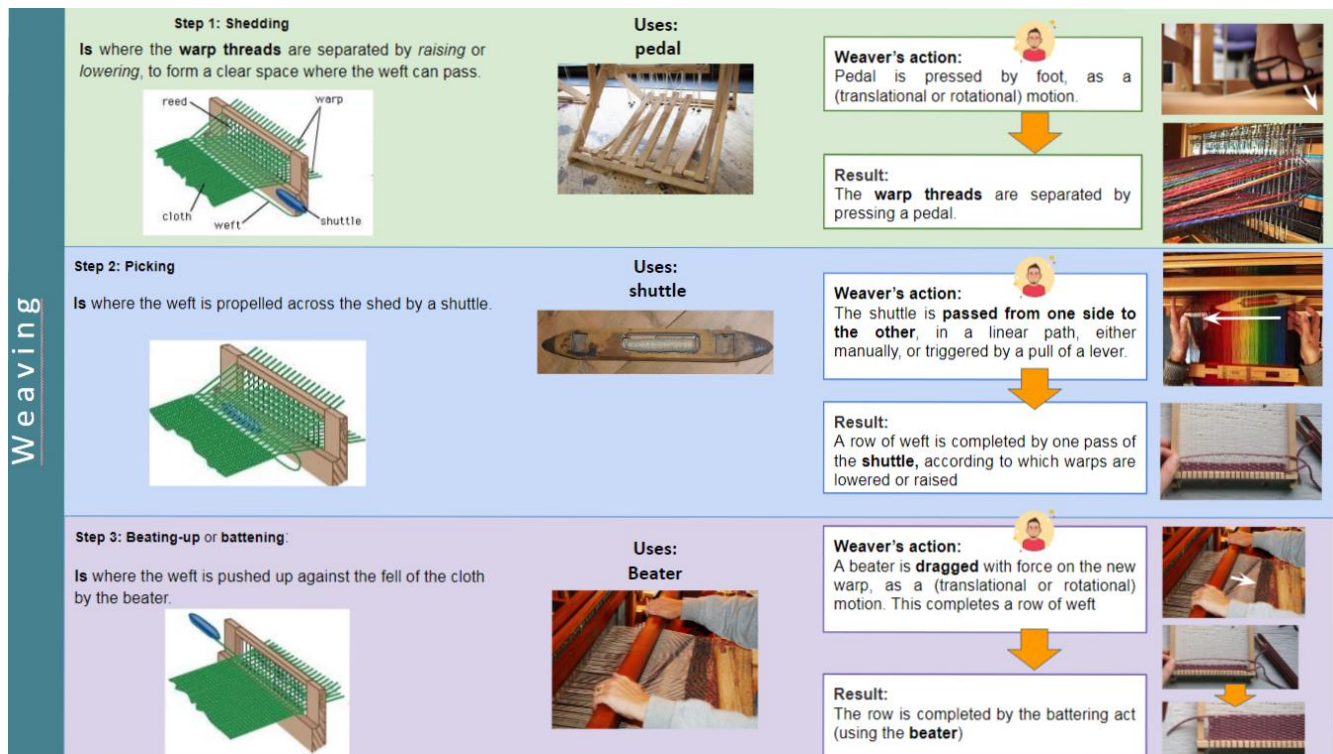


Figure 14: Overview of the weaving process: steps, actions, and FMCs involved. (source: Mingei, 2020) [95]

5 Application of the proposed methodology to the two pilot studies

5.1 Application of the Proposed Methodology for Handheld Tools: TooltY

This Chapter describes how we applied the proposed methodology for the presentation of crafts involving handheld tools in Virtual Environments, utilizing Virtual Humans as the practitioners. An integral part is the correct attachment of the tools to the VH's hands, as well as the induced motion of the tools from the human motion. To that end, and as a first approach into investigating these matters, TooltY [82] was developed, focusing on simple handicrafts for handheld tools. As an example, we showcase how the proposed methodology was applied for the use case of a VH operating a hammer.

TooltY's pipeline is presented (Figure 15), from the Motion Capture of the human movement, to an authored 3D scene which includes the Virtual Human(s), the tools, and various 3D objects, as well as the motion of the VHs and the tool they use. At the beginning, this scene is empty, and subsequently the VH, the tool, and finally the surrounding environment (room and 3D objects) are added.

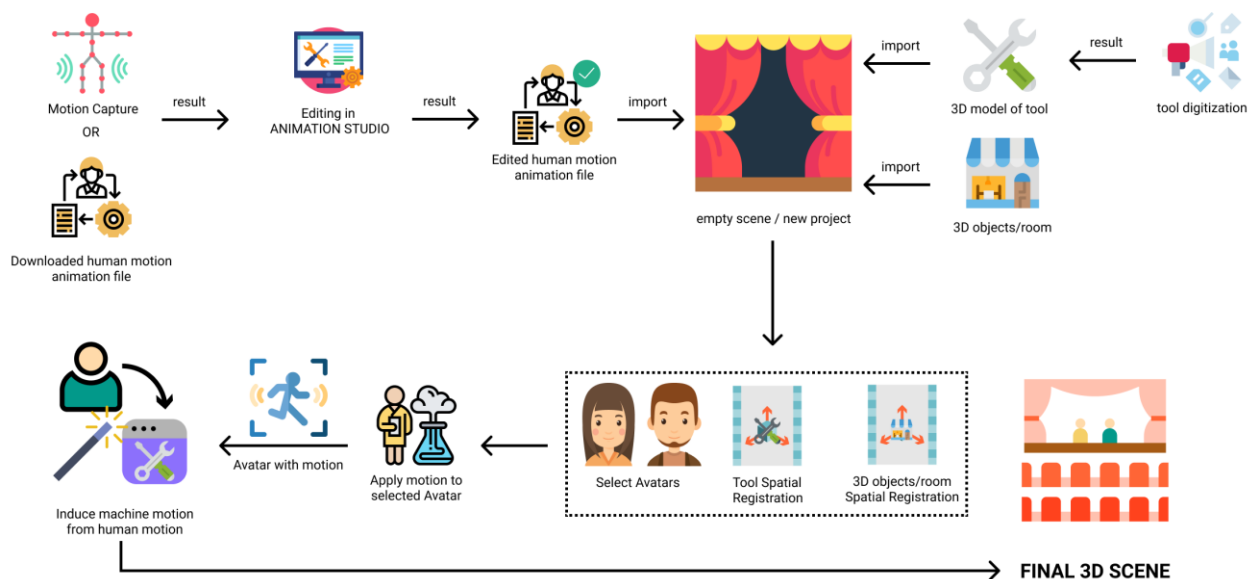


Figure 15: Overview of TooltY's pipeline. (source: Mingei, 2020) [96]

5.1.1 Step 1: Animation File for Human Motion

The first requirement in the pipeline is an animation file, which represents the human motion while operating the tool. This can either be the result of Motion Capture, or also any appropriate animation (e.g., procured from the web) can be used. For the hammering activity explored in the context of TooltY, we used an open BVH dataset from outworldz⁵, and specifically package 60-75⁶ file 62_08.bvh. The Biovision Hierarchy (BVH) is a character animation file format, and is the resulting animation file when conducting Motion Capture. In our case, the BVH animation file was

⁵ <https://www.outworldz.com/Secondlife/Posts/CMU/>

⁶ <https://www.outworldz.com/Secondlife/Posts/CMU/cmuconvert-daz-60-75.zip>

converted to a FBX animation file, for use in the pipeline, since TooltY was developed in Unity 3D, which supports FBX animation files and not BVH.

5.1.2 Step 2: Tool Digitization

Another prerequisite concerns the digitized form of the tool(s) that will be used. For this, various approaches for 3D reconstruction for digitization can be used as presented in depth in D1.3. The result of this digitization is a 3D model that represents the surface geometry and appearance of the object and is, typically, encoded as a textured mesh of triangles (e.g., in VRML or FVB file format). While we can use the model of any digitized tool, some cases require post-processing. For instance, in the case of scissors, post processing is needed to make the tool deformable, in order to introduce joints that describe its deformation. In the context of experiments with TooltY, several 3D models were used, coming both from 3D reconstructions of physical objects and from online sources⁷. For complementarity, in the context of TooltY we explored the use of both non-deformable (i.e., hammer) and deformable (i.e., scissors) tools.

5.1.3 Step 3: Editing of Animation Files in Animation Studio

For editing the animation files, we use a BHV editor, developed in-house, called Animation Studio (AnimIO presented in D1.3 and D5.1). Animation Studio allows visualization, editing, and annotation of 3D animation files in BVH format (obtained by motion capture or visual tracking). In the case of visual tracking (presented in D5.4), a 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, but does not stream for video and motion capture. Using Animation Studio, segments of the BVH animation file can be isolated and exported to test different “atomic” scenarios (e.g., hand movement when hammering a nail in a more complex process, involving hammering other objects), simplify the input, or allow the in depth analysis of certain scenarios.

5.1.4 Step 4: Start TooltY

When starting TooltY, the first step is to create a new project, which loads an empty scene. After editing the animation files, they can be imported to TooltY, along with the 3D models for tools and any desired objects. Users can then utilize these imported files in their scenes.

5.1.5 Step 5: Selecting an Avatar for the Virtual Human

The Avatar representing the Virtual Human plays a very important role in this context, as it is the actor executing the movements representing the tool usage, thus bringing the whole process to life. The user can choose from a selection of available Avatars. TooltY is built using the Unity 3D game engine, and thus Avatars created using a plethora of 3D Computer Graphics and Animation Creation editors can be imported (e.g., 3DStudioMax, Maya, etc.). During the development of TooltY, Poser Pro 11⁸ as well as Adobe Fuse⁹ were employed for the creation of the two (2) Avatars

⁷ www.thingiverse.com

⁸ www.posersoftware.com

⁹ https://www.adobe.com/gr_en/products/fuse.html

visible in Figure 16. These Avatars were exported from Poser and Fuse, and then imported to the developed platform as resources that can be selected and assigned to a simulation scenario. After an Avatar for the Virtual Human has been selected, it can be added to the scene.



Figure 16: Screenshot of the 2 Virtual Humans holding (a) a hammer and (b) scissors. (Source: Mingei, 2020) [97]

5.1.6 Step 6: Application of Motion to the Virtual Human

After a suitable Avatar has been chosen, the user can apply on it a single or multiple human motion animations. Each animation is mapped through the correspondence of the joints between the humanoid-type rig (skeleton) of the Avatar, and the joints of the humanoid skeleton of the animation file. The result of this process can be previewed both using the selected Avatar and in the form of a primitive Avatar animation. The latter is used to help users visualize animation problems (e.g. elbows are getting inside the body when movement is previewed using a certain avatar). This could mean for example, that the avatar's torso is too narrow. This is a known and well-studied problem in Computer Graphics, known as motion retargeting, and a lot of research work has focused on solutions, such as [83], [84] and [85]. In the scope of this research work, we decided to solve the problem offline, by utilizing Unity's Avatar Muscle & Settings¹⁰, for configuration of the degrees of freedom of joints in the skeleton of the Avatar. In more detail, this module allows tweaking of the character's range of motion to ensure the character deforms in a convincing way, free from visual artifacts or self-overlaps. However, we would like to explore other

¹⁰ <https://docs.unity3d.com/Manual/MuscleDefinitions.html>

possibilities for online motion retargeting in future work, in order to minimize the manual and offline tweaking required. An example of a Virtual Human operating a hammer can be seen below, in Figure 17.



Figure 17: Screenshots of a Virtual Human holding and operating a hammer. (Source: Mingei, 2020) [98]

5.1.7 Step 7: Application of Motion to the Virtual Human

The attachment of the handheld tool to the Virtual Human’s hands is a very important part of the process. Below our methodology to implement this is presented:

The digitized object of the tool o is represented by a mesh of triangles. As was explained in the section presenting the proposed methodology for handheld tools, the following are required for introducing a tool:

- A grip point p_a on the hand of the Avatar.
- A preferred grip on the tool, denoted as g .
- Two points on the front and back faces of the desired grip point of the tool, denoted as p_f and p_b , respectively.
- A grip center p_c , which is automatically calculated as the mid-point of p_f and p_b .
- The projection of p_c to the top side of the bounding box of the tool o , denoted as p_g , also automatically calculated.

Below, translations are encoded as 3x1 matrices and rotations as 3x3 rotation matrices. The coordinate frame of the hand C_h is the coordinate system on the selected joint of the Virtual Human (e.g., in the case of operating a hammer, the VH’s right hand). This frame is determined, at each moment in time, by the animation that the VH executes.

A grip g is a configuration of the VH’s hand that is represented by a series of rotations of the VH’s joints. The number of rotations is determined by the skeleton of the VH and the values of these rotations by the animation that it executes.

A tool posture is the rotation that orients the tool in the hand of the VH as intended by this posture. The posture is represented by rotation matrix R . Matrix R aligns the C_h with C_o , by rotating

the tool in-place, by \mathbf{R} . The rotation center is \mathbf{p}_c , and \mathbf{R} needs to be an “in place” rotation to avoid rotation about the world center. \mathbf{R} is determined by the current orientation of the hand. It should be noted that we choose to rotate around \mathbf{p}_c , since that is the point from which the VH holds the tool.

Let \mathbf{b}_o the bounding box of tool \mathbf{o} . We need to name the faces of \mathbf{b}_o as front / back, top / bottom, and left / right. This determines the coordinate frame \mathbf{C}_o for tool \mathbf{o} . This frame is represented by the rotation matrix \mathbf{R}_o that aligns the \mathbf{C}_o to the world coordinate frame.

Typically, the VH and tool 3D model are not in the same scale, metric unit, and coordinate system. Tool \mathbf{o} is brought to the correct scale by scaled by factor \mathbf{s} ; the scaled tool is denoted as \mathbf{o}_s . Let \mathbf{s} the scalar that adjusts the scale and metric unit of the model. To emulate the grip of a tool, grip \mathbf{g} is applied to the VH. The transformation that brings the tool in the hand of the VH is as follows:

Let \mathbf{x} a 3D point of \mathbf{o} .

Let $\mathbf{T} = \mathbf{p}_a - \mathbf{p}_c$ the translation that brings \mathbf{p}_c and \mathbf{p}_a to coincidence.

The required (see above) in-place rotation is applied to \mathbf{o} about \mathbf{p}_c . The operation required is $(\mathbf{R} * (\mathbf{x} - \mathbf{p}_c)) + \mathbf{p}_c$, where ‘*’ denotes matrix multiplication.

The above-presented formulas are described for completeness and generalization purposes of the methodology. Since our development platform is Unity, it should also be noted that the Unity Engine internally maintains a hierarchy-based (parent-child) transformation matrix stack. However, it does not allow for the direct manipulation of this matrix stack, but instead provides access to some exposed properties in order to perform transformations by code/scripting. These exposed properties include the position in the form of a 3D Vector, the rotation in the form of a Quaternion, and scaling in the form of a 3D Vector. Thus, in the scope of developing TooltY inside the Unity 3D environment, while the definition of the grip points and the application of translation and rotation are performed via the aforementioned exposed properties, to achieve the correct attachment of the tool to the hand, it is important to note that some of the aforementioned formulas, including the scaling, were implemented via Unity’s Engine.

It should also be noted that in Unity, the default pivot is the center of the mesh of the object, and it is according to this point that any transformation is applied on the object. Therefore, in order to manipulate the object around \mathbf{p}_c , we took advantage of the parent-child relationship in Unity, and the fact that this is provided out-of-the-box, and created an intermediate (invisible) object, as a parent of the tool in the hierarchy, to act as the custom pivot. Provided we set the custom pivot parent object in the correct world space coordinates, the tool object is then manipulated by local offsets, based on the pivot parent object. As a result, transformations on the pivot object will affect the tool object, via the parent-child relationship, as expected. Figure 18 below aims to explain the aforementioned concepts, notations and operations.

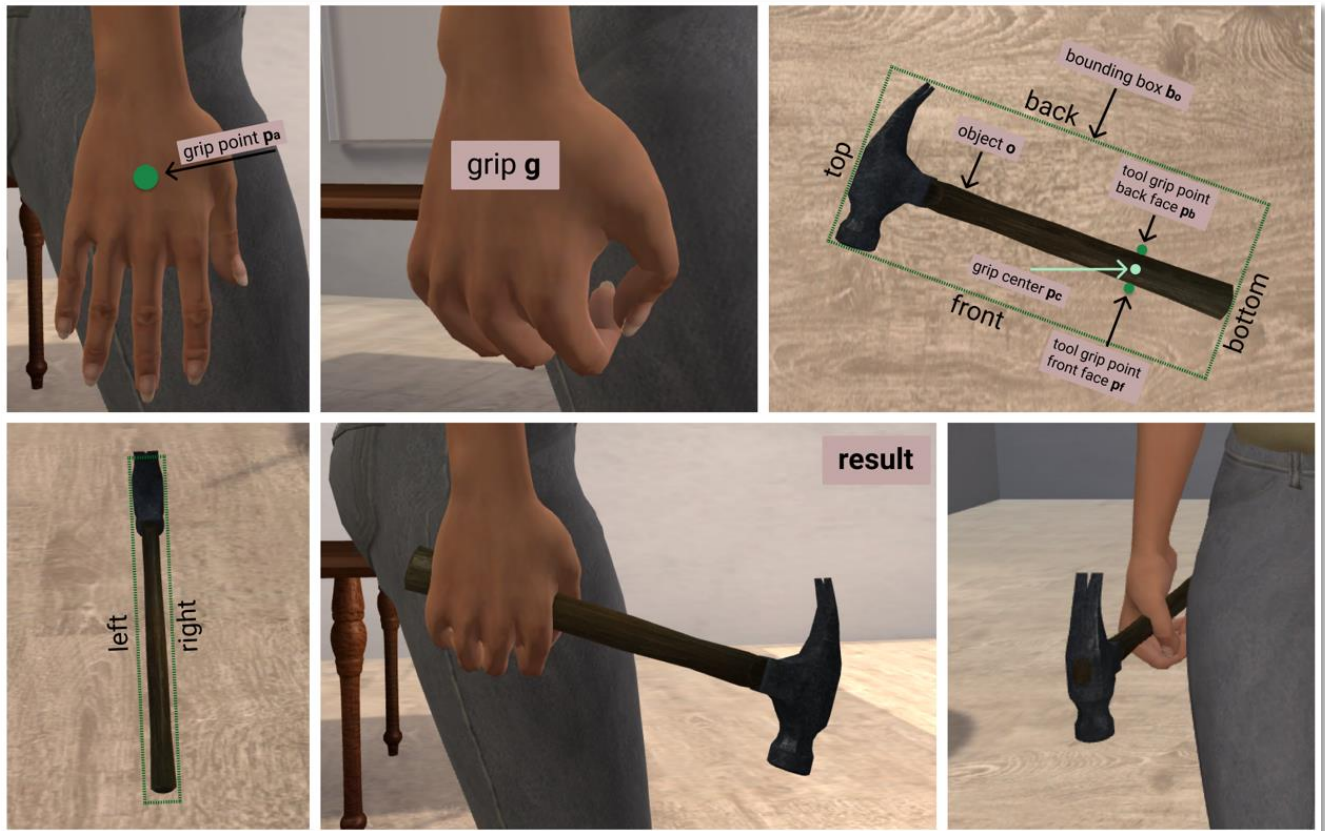


Figure 18: Grip points, orientations and resulting attachment of hammer to the VH hand. (Source: Mingei, 2020) [99]

5.1.8 Step 8: Addition and Spatial Registration of Room/3D Objects

The user can also add “decorative” 3D objects to the scene. Such objects can either be craft workspace objects (i.e., pieces of furniture) or the room itself, i.e. the 3D environment where the scene is set. As with the 3D models of the tools, the room and the objects imported may derive from heterogeneous sources and models, and thus spatial registration may need to take place, i.e. the 3D models may need to be scaled, rotated or translated.

5.1.9 Step 9: Tool Manipulation from Human Motion

The proposed methodology does not use Motion Capture animation files for the movement of the tools, so the tool motion is induced from the human motion. We argue that it is easier and more cost-efficient to induce the tool motion, since the tool may not be rigid (e.g. scissors) or/and might not be usable if we add motion capture instrumentation on it. For example, if sensors for motion capturing were put on a scissor, it might not be possible for the operator to hold it and use it easily or as they normally would. Our approach aims to solve such issues, by allowing generalization, as we can induce the motion of any simple handheld tool, by that of humans.

As explained in Chapter 4, the tool manipulation methodology depends on whether the tool is deformable or not. In the case of the non-deformable hammer, which was the demo-case that the development of TooltY mainly focused on, its movement derived from applying transformation

operations on the tool object, according to the transformation of the hand operating it, as it is moving in the Virtual Environment based on the animation file.

5.1.10 Step 10: Play the Scene in 3D

After following these steps, the user can choose to play the scene, i.e. trigger the execution of the animations that comprise the MV. Thus, the VH comes to life, reenacting the use of the tool, in the desired 3D environment. Namely, TooltY produces Virtual Environments where the VH demonstrates how to execute a handicraft, step by step.

5.2 Application of the Proposed Methodology for the Loom Weaving Case

The following sections present the foundation and theory behind the implementation of the proposed approach for craft visualization for the case of loom weaving [86]. This entails the application of the proposed methodology, as discussed in Chapter 3.

5.2.1 Motion Capture

As discussed in the Related Work section, there are various ways to produce Motion Capture animation files, and all of them are compatible with the proposed approach. **In the context of Mingei and for the example of loom weaving, MoCap files that were the product of motion capturing of the practitioners at Haus der Seidenkultur in Krefeld Germany are used.** This was conducted with a NANSENSE© R2¹¹ motion capture suit, in the context of MoCap sessions that took place during the second plenary meeting of the Mingei.

5.2.2 Loom Model Acquisition

Often the machine is extremely difficult to reconstruct, due to its placement in the constrained environments of workshops and museums. As this was the case with the looms we had at hand at HdS, we used a basic loom model¹² to demonstrate our approach. Of course, any 3D model of the machine could be used.

5.2.3 Virtual Humans

The VHs that reproduce the recorded actions are 3D Avatars. For MoViz we created different Avatars using Poser Pro 11 as well as Adobe Fuse, and then imported them to our development platform (*Unity3D*).

5.2.4 Motion Vocabulary

The next step entails the decomposition of the process of loom weaving into actions, as discussed in the corresponding section is Chapter 4. Thus, we edited the MoCap animation files to correlate the motion segments (MVIs) to the conceptually decomposed actions, which are (i) shedding, (ii) picking and (iii) battening.

¹¹ <https://www.nansense.com/>

¹² <https://3dwarehouse.sketchup.com/model/a4d5115a90e3f5534cf6cee9a1fdf035/Counterbalance-Loom>

5.2.5 Loom Machine Abstraction

We then proceeded with defining the elements of the physical interface of the loom as FMCs. Each one is comprised of (i) a 3D model of a machine part, and (ii) motion rules that represent the feasible, induced motion of the FMC during its operation. The three (3) FMCs in the case of loom weaving are the (i) treadle, (ii) shuttle and (iii) beater, corresponding to the aforementioned motions of (i) shedding, (ii) picking and (iii) battening.

Depending on the acquired 3D model for the machine, some additional editing might need to take place. That is, in case the identified FMCs are “attached” to the 3D object of the machine, they need to be separated in some 3D editing or modeling tool. In the case of the loom machine we used, the Maya 3D modeling software was used to separate the mesh of the machine into (i) the basic loom model, and two (2) of the three (3) FMCs: the (ii) treadle and (iv) beater. The specific loom machine did not contain a shuttle, so we imported one resulting from the 3D reconstruction conducted at HdS, Krefeld, along with the MoCap of the practitioners. Moreover, for the treadle and the beater, additional editing included the addition of joints to their 3D models, in order to achieve the appropriate motion rules for each of them. In more detail, the treadle needs to behave like a hinge joint, i.e., attached at one point and performing a rotational movement according to the force applied on its other side. On the other hand, the beater has two (2) points of attachment to the base loom machine, and needs to also perform a rotational motion, with these two (2) points never moving. To facilitate understanding, Figures 47 and 48 below show these joints and the motion that these 2 FMCs should perform.

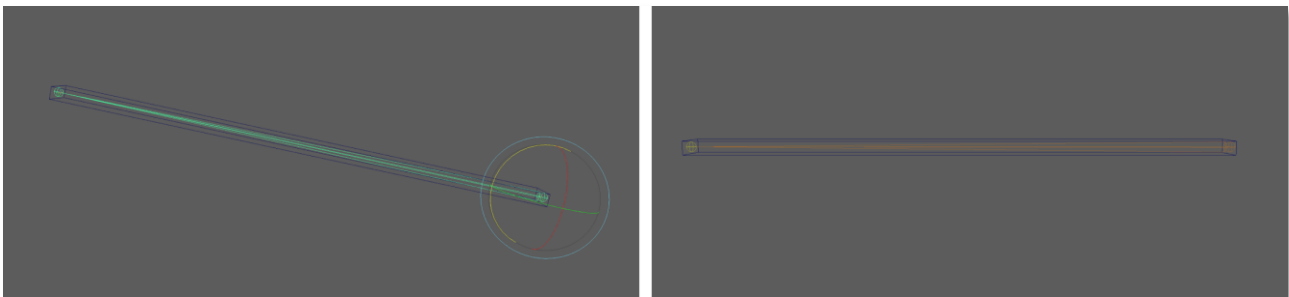


Figure 19: Loom treadle model in its “max” and “min” positions, with joints visible. (Source: Mingei, 2020) [100]

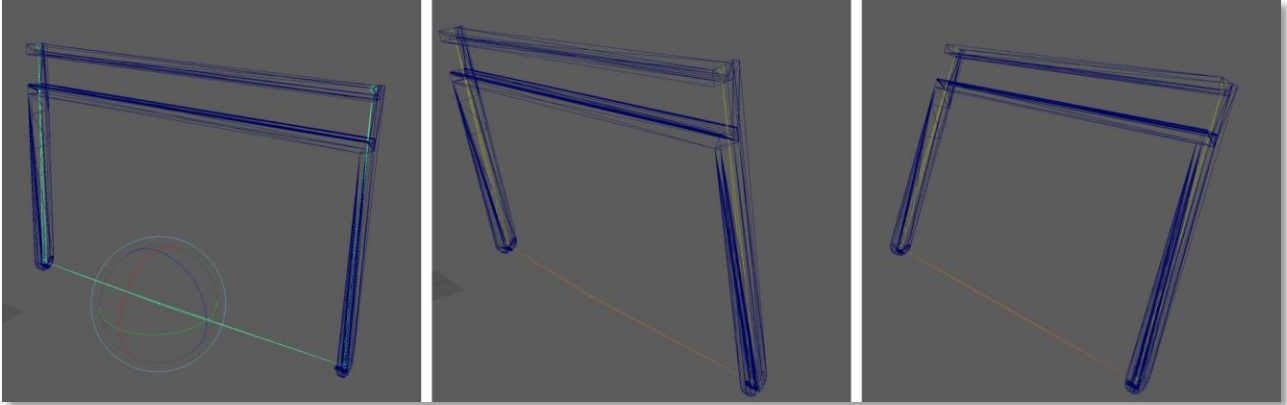


Figure 20: Loom beater model in its idle state and “max”, “min” positions, with joints visible. (source: Mingei, 2020) [101]

Finally, the treadle of the loom model we downloaded was incompatible with the one of the real loom machine used during the MoCap sessions. Therefore, Maya was once again used in order to rotate the pedal and bring it in a desirable position. This is clearly depicted in Figure 49, where the original model is compared with the resulting one.

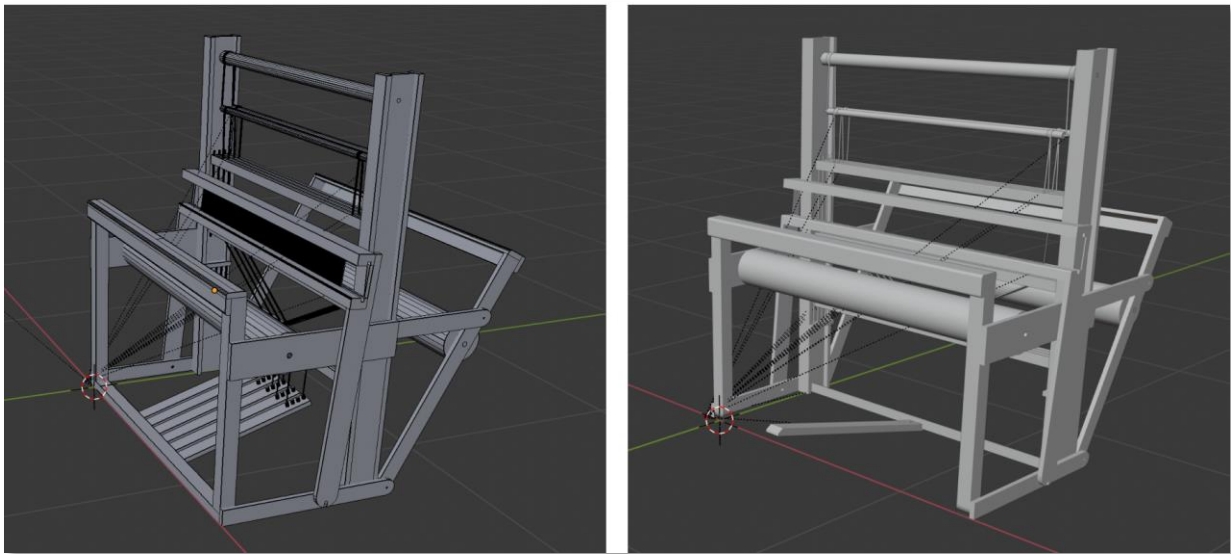


Figure 21: Downloaded loom model vs model after editing (noticeable difference in the treadle mechanism) (source: Mingei, 2020) [102]

5.2.6 Association of Virtual Humans and FMCs

The loom machine interface components, represented as FMCs, are associated with the VH body part(s) that are used for their operation (e.g., treadle with foot, shuttle with right hand, beater with left hand). We first establish a pairing between the FMC and a point on the Avatar. Subsequently, the preferred grip posture is defined. For this purpose, we employ the following entities:

- Avatar **A**, with skeleton **S**, is comprised of joints and skin **T**. Skin **T** is a, possibly textured, deformable 3D surface, represented by a mesh of triangles. When **A** is animated, **T** deforms according to the motion of **S**.
- Two grip points, $\mathbf{g}_L, \mathbf{g}_R$ on **T** encode the grip center for the left and right hand respectively. These points are selected with respect to the FMC, as objects may not always be held in the same way.
- Each hand has a reference frame based on orthogonal unit vectors. These are u_x^L, u_y^L, u_z^L , for the left and u_x^R, u_y^R, u_z^R for the right. The center of the frame is selected at an anatomically meaningful location.
- An FMC has a reference frame based on u_x^M, u_y^M, u_z^M , which are orthogonal unit vectors for the FMC. Each FMC is cantered at its centroid.
- The FMC has a preferred usage position (e.g., grip, foot position). This position may not be unique.
- A posture **p** is comprised of (i) a configuration of the joints of **A**, (ii) a preferred location and (iii) orientation of **A**'s body members, for FMC usage.
- An animation is a transition from a posture to another, represented by a sequence of states of the **A**'s joints.

A **preemptive posture** is the preferred for **A** posture at the first moment of the FMC usage. A **preemptive animation** is an animation that brings **A** to the preemptive posture.

Furthermore, we define the following concepts:

- Loom **L** is represented by a mesh of triangles, encoded by its vertices, \mathbf{l}_v , and its triangles, \mathbf{l}_t .
- TRE, BEA, SHU** are FMCs for the treadle, beater and shuttle, respectively.
- Points \mathbf{b}_L and \mathbf{b}_R on the **BEA**, denote the grip locations of the left and right hands.
- Animations \mathbf{p}_L and \mathbf{p}_R , for preemptive usage postures for the **BEA**, for the left and right hand.
- Preemptive usage animations \mathbf{f}_L and \mathbf{f}_R for the placement of the left and right feet on **TRE**.
- Point \mathbf{d}_a on **TRE** at the center of the area that the foot is pressing on the treadle. Preemptive usage postures \mathbf{s}_L and \mathbf{s}_R encode shuttle grip by the left and right hand.
- Point \mathbf{u}_s on **SHU** (shuttle centroid).
- MV for Loom Weaving \mathbf{MV}_{LW} contains \mathbf{MVI}_{TRE} , \mathbf{MVI}_{BEA} and \mathbf{MVI}_{SHU} which are the treadle, beater and shuttle animations (encoding human motion but not machine motion):
 - \mathbf{MVI}_{TRE} : treadle pushed down and released.
 - \mathbf{MVI}_{SHU} : shuttle pushed from left to the right and vice versa.
 - \mathbf{MVI}_{BEA} : beater pulled towards the operator for a beat, then pushed away.
- Animation function** $\mathbf{AN}(\mathbf{A}/\mathbf{FMC}, \text{Posture})$ which animates either the **A** or FMC according to a MVI.

A scene is the VE where the **A**, FMCs and objects are instantiated, and the FMCs, **L**, and **A** are brought to the reference frame of the scene. This is achieved by an appropriate **rotation** **R** (encoded as a 3x3 rotation matrix), a **translation** **t** (encoded as a 3x1 matrix) and a **scaling** **s** transformation for each component, once and prior to their import in the VE. Each 3D point, let **q**, of the object's 3D model undergoes transformation $\mathbf{R}^* \mathbf{s} \mathbf{q} + \mathbf{t}$, where $*$ denotes matrix multiplication. The transformations are individual for each of the FMCs, loom and **A**, and are denoted as $\mathbf{R}_T, \mathbf{s}_T, \mathbf{t}_T$ for the treadle, $\mathbf{R}_S, \mathbf{s}_S, \mathbf{t}_S$ for the shuttle etc.

5.2.7 Induced Machine Motion

Induced machine motion can be modeled as follows: **Principle of induced motion.** Let avatar **A** at the preemptive usage posture of an FMC. We consider the execution of a **MVI** by **A**, during a time interval. The motion of the FMC due to the MVI is called **Induced Motion** of the FMC. We propose a synchronization method of the FMC's motion with that of the VH for each MVI, based on the feasible induced motion trajectory of the FMC (result visible in Figure 53).

5.2.7.1 Treadle motion

Treadle motion is performed by execution of MVI_{TRE} and denoted as $AN(A, MVI_{TRE})$. The treadle is moved when the bounding box of the **TRE** collides with the foot of the Avatar. The virtual motion is achieved through a function $TRE' = AN(TRE, MVI_{TRE}')$, where MVI_{TRE}' contains the projections of MVI_{TRE} to the motion trajectory of **TRE** (Figure 50).

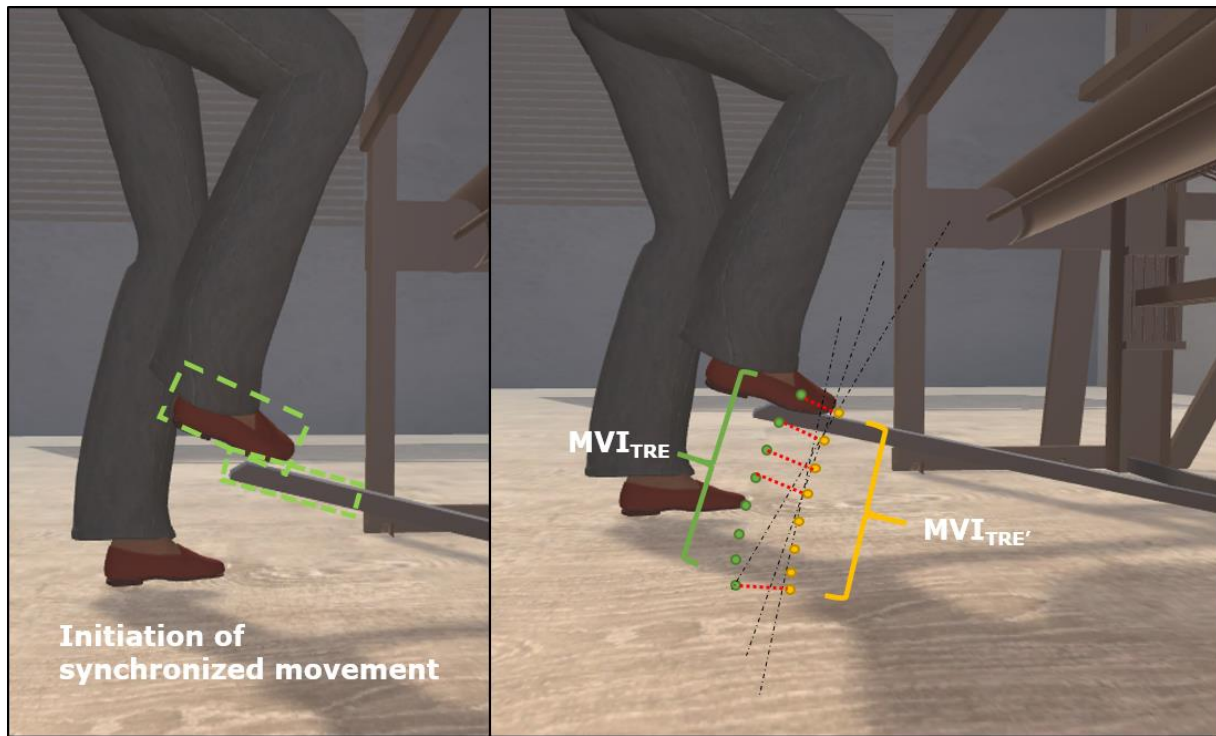


Figure 22: Visualization of the foot pressing the treadle. (source: Mingei, 2020) [103]

5.2.7.2 Beater motion

Beater motion. The preferred posture of **A**'s hands is reached by animating **A** using a preemptive animation, so that $A' = AN(A, p_L)$, $A' = AN(A, p_R)$. Hand motion is performed by MVI_{BEA}' through function $A' = AN(A, MVI_{BEA}')$ and loom motion through $L' = AN(L, MVI_{BEA}')$, where MVI_{BEA}' contains the projections of the grip points to the motion trajectory of **BEA** (Figure 51).

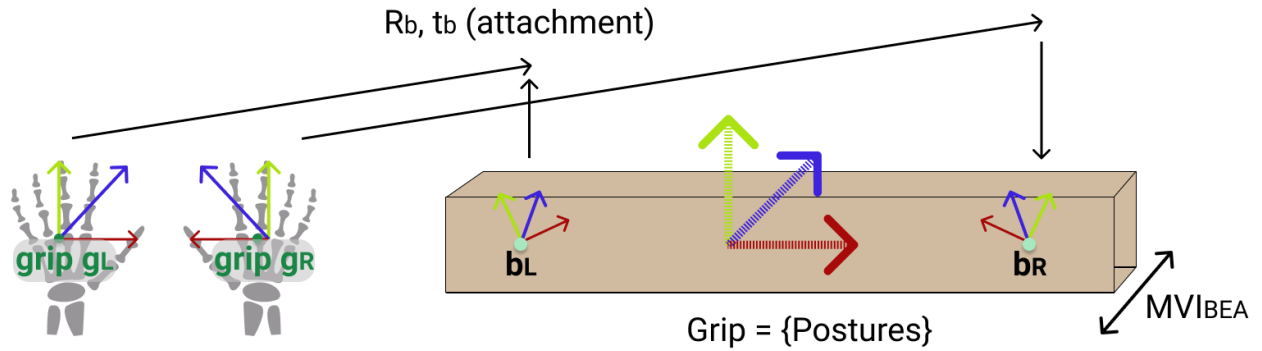


Figure 23: Attaching the hands on the beater. (source: Mingei, 2020) [104]

5.2.7.3 Shuttle motion

Shuttle motion. Shuttle motion MVI_{SHU} is simulated through function $A' = AN(A, MVI_{SHU})$, while the attachment of the shuttle to each of the hands of the Avatar is performed by $A' = AN(A', s_L)$ and $A' = AN(A', s_R)$ for the left and right hands respectively (shuttle is exchanged between the hands) (Figure 52). The motion of the shuttle is represented as $L' = AN(L, MVI_{SHU})$, where MVI_{SHU}' is modeled as a constant linear motion, between the starting point of the **SHU** feasible induced motion trajectory and its ending point, and vice versa.

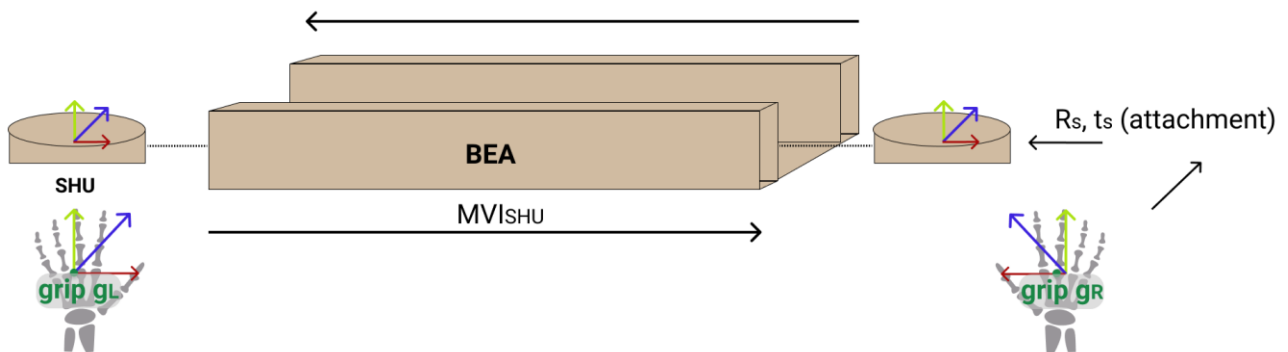


Figure 24: Attaching the hands on the shuttle. (source: Mingei, 2020) [105]



Figure 25: Visualization of the result: the VH is operating the loom. (source: Mingei, 2020) [106]

6 Implementing an Authoring, Visualization and Training Platform for Craft Experiences (MoViz)

TooltY mainly focused on the implementation (i) of the correct attachment of the handheld tools to the body parts operating them and (ii) of the motion of the tool involved in the process. At that point, no User Interface (UI) was yet designed; the final Mingei UI and the complete Interaction Paradigm were developed for MoViz. To that end, prototypes were designed according to the User Requirements, followed by expert based evaluations to verify their usability, before moving on with the implementation as presented in D4.2.

6.1 Requirements for the MoViz platform

As the second and more complete application of the proposed methodology, discussed in Chapter 3, the MoViz platform was developed, supporting the authoring, demonstration and training of craft usage experiences, which include not only handheld tools, but also machines. It encompasses two (2) tools: The Motion and Scene Editor (MSE), and the Motion and Scene Player (MSP). Below, the high-level functional requirements that MSE and MSP satisfy are presented, which have been

collected through an extensive literature review and iterative elicitation process, based on multiple collection methods such as brainstorming, focus groups, observation and scenario building.

MSE = Motion and Scene Editor - Users should be able to:

1. Load existing scenes, choose to create a new one, and save their progress.
2. View a collection of available items to add to the scene (Avatars, Motions, Fundamental Machine Components, Scene Objects).
3. Be able to view the 3D items that have been added to the scene.
4. Modify the 3D items added to the scene. In more detail:
5. Add an Avatar **A** (Virtual Human) to the scene as the operator. A number of alternative Avatars should be supported (male, female, short, tall, fat, slim, etc.). The selection of the Avatar is important in order to (a) enhance realism, (b) be flexible in the representation of gender issues (e.g., in Greece looms are mainly operated by women while in Germany we have both male and female experts).
6. Add Motion Vocabulary Items (MVIs) to the Avatar.
7. Add a motion to the scene (associate a motion with an MVI of the Avatar).
8. View which motion has been assigned to which MVI of the Avatar.
9. Add an FMC to the scene (associate an FMC with an MVI of an Avatar).
10. View which FMC has been assigned to which MVI of the Avatar.
11. Add a Scene Object from the collection to the scene.
12. Delete an MVI associated with an Avatar.
13. Remove the FMC or Motion assigned to an MVI.
14. Export the scene (Avatar with corresponding MVIs and the “room”, i.e. any Scene Objects added).
15. Preview each MVI, or the entire MV (i.e. play the motion).

MSP = Motion and Scene Player - Users should be able to:

1. Load a scene (exported from MSE).
2. Set the order of the MVIs, i.e. configure the order in which the animations are played, and therefore the resulting MV.
3. Choose to *play* the scene, i.e. play the list of motion vocabulary items in the scene.
4. Before playing, configure the playback:
 - a. Configure the type of playback for each Avatar (3D virtual environment/VR).
 - b. In the case of VR, activate or deactivate training mode, where the ideal trajectories should be visualized, and the user trajectories should be recorded for comparison with the ideal ones.

6.2 A Look at the Resulting Platform’s UI

MoViz underwent two (2) expert based evaluations as presented on D4.2 and based on the results the MoViz platform prototype was redesigned and implemented. The following Figures 30-41 visualize some of the most important screens of the platform as examples.

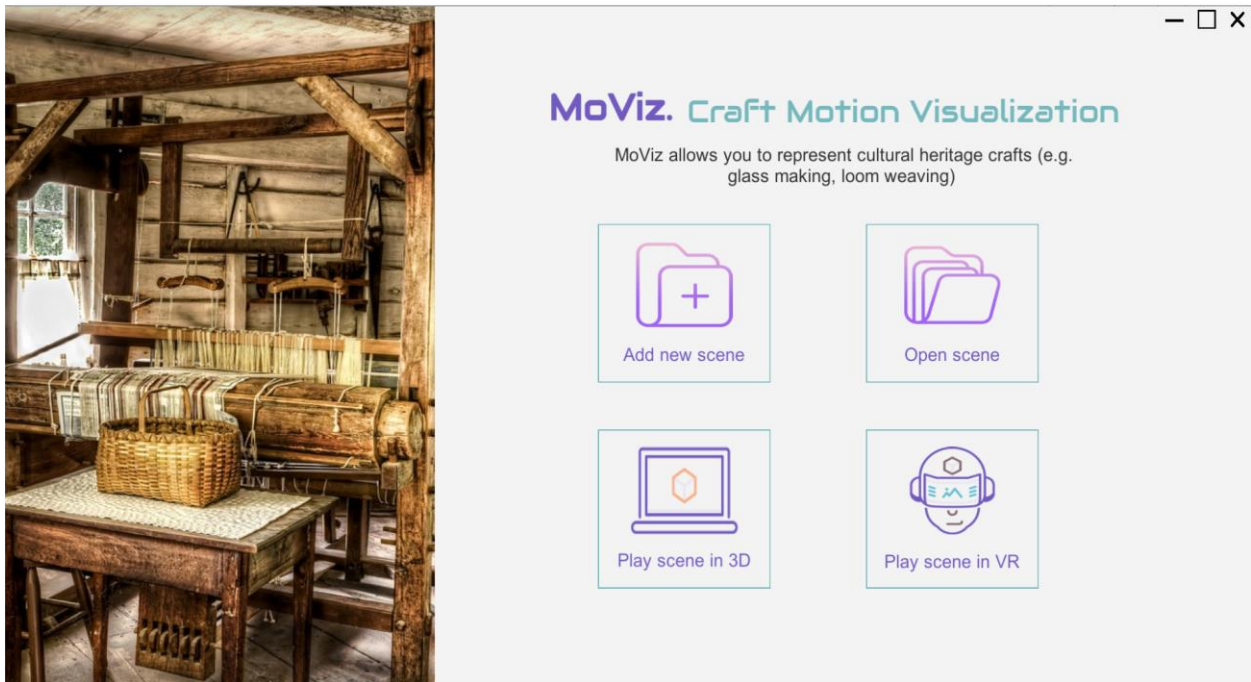


Figure 26: Design 3.0 - MoViz Start Screen. (source: Mingei, 2020) [107]

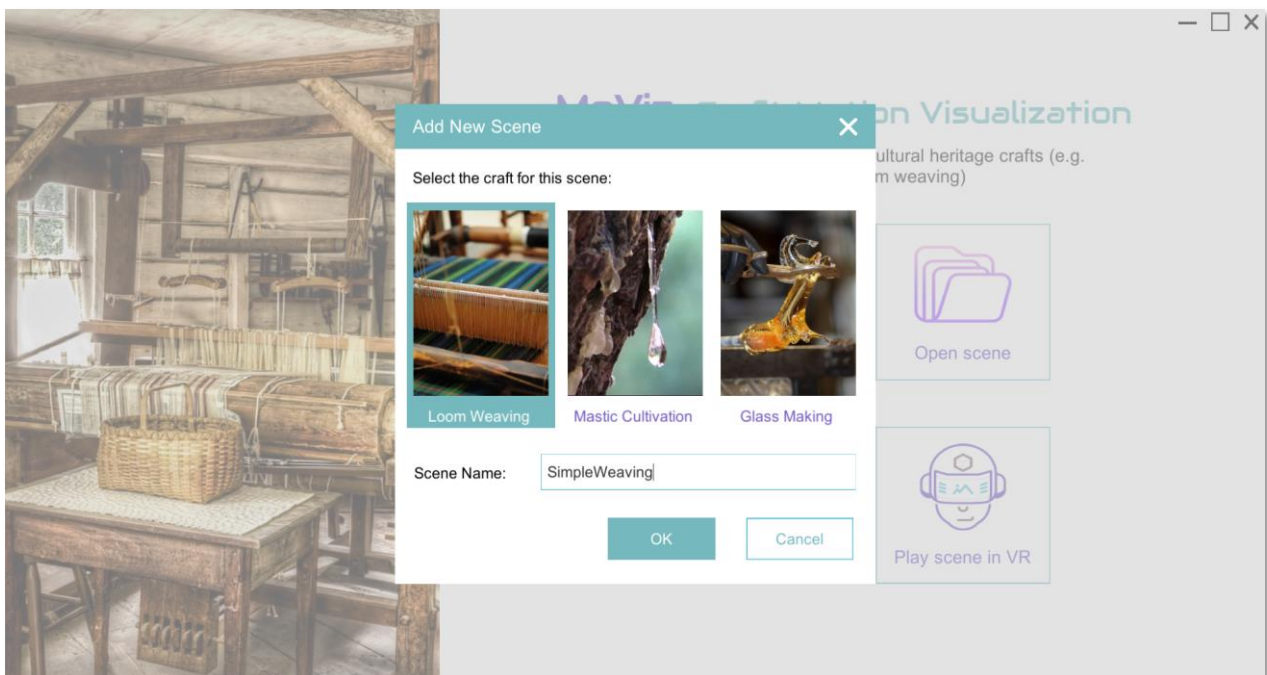


Figure 27: Design 3.0 - MoViz Start Screen - Add New Scene Selected. (Source: Mingei, 2020) [108]

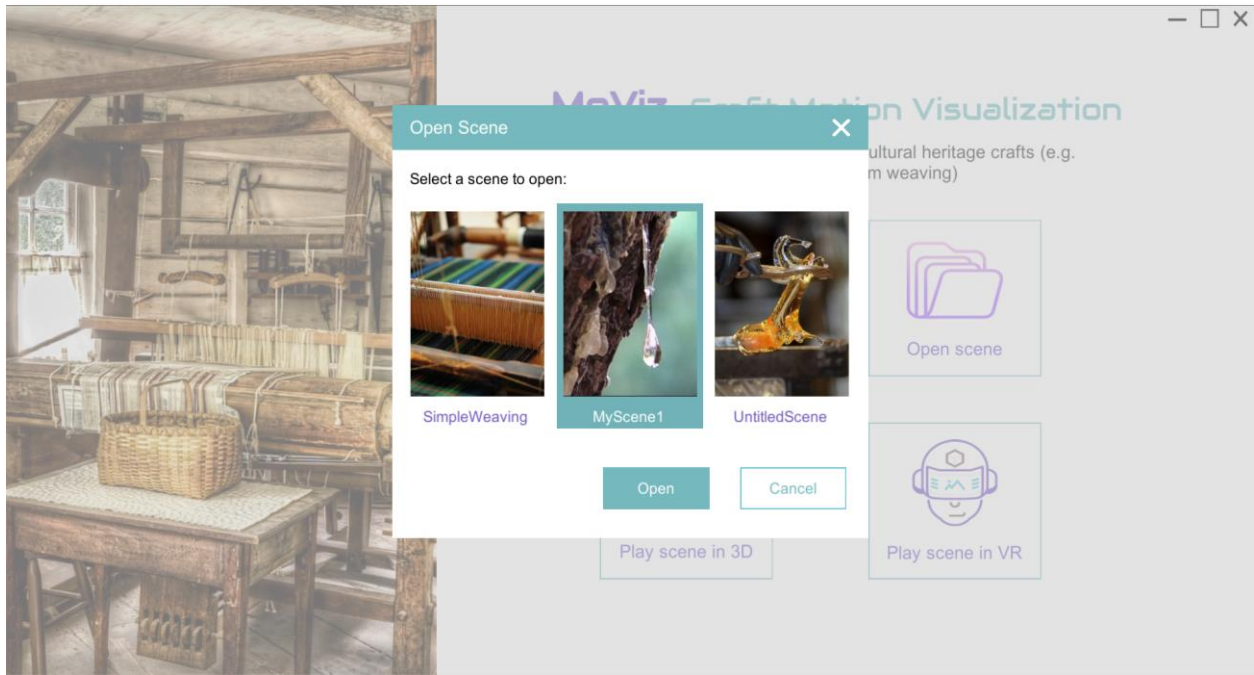


Figure 28: Design 3.0 - MoViz Start Screen - Open Scene Selected. (source: Mingei, 2020) [109]



Figure 29: Design 3.0 - MSE - Empty Scene. (source: Mingei, 2020) [110]

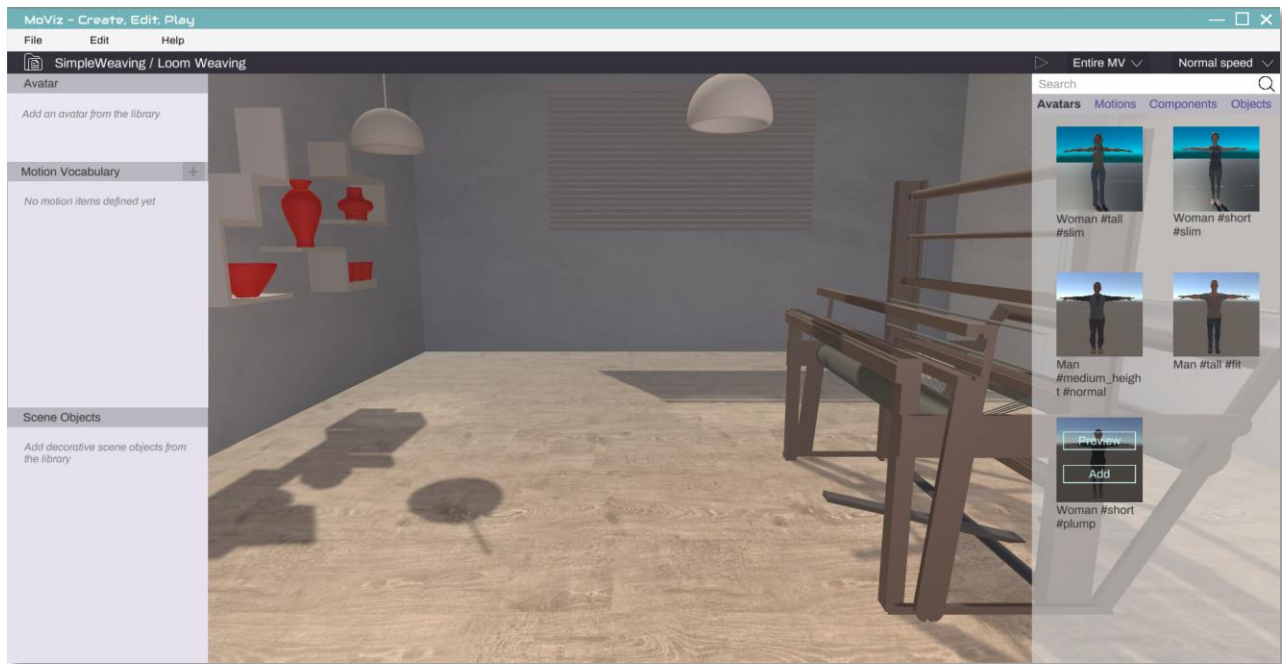


Figure 30: Design 3.0 - MSE - empty Scene, hovering over Avatar in Library. (source: Mingei, 2020) [111]

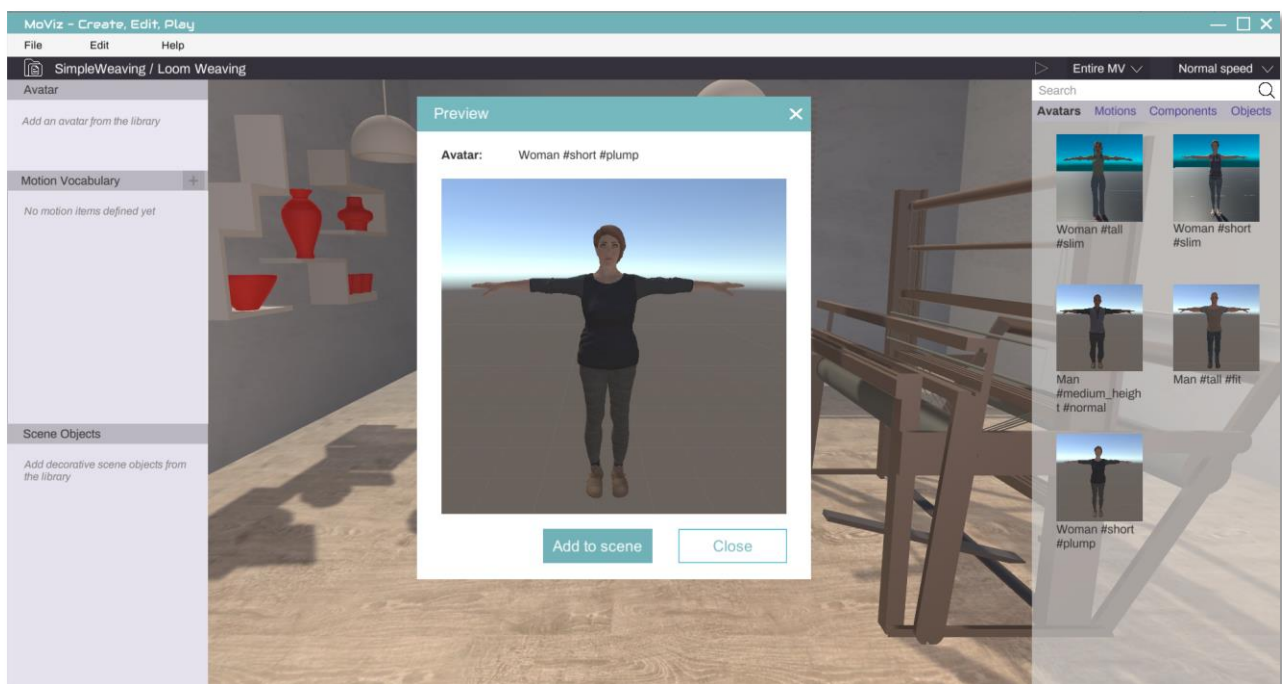


Figure 31: Design 3.0 - MSE - Avatar Preview. (source: Mingei, 2020) [112]

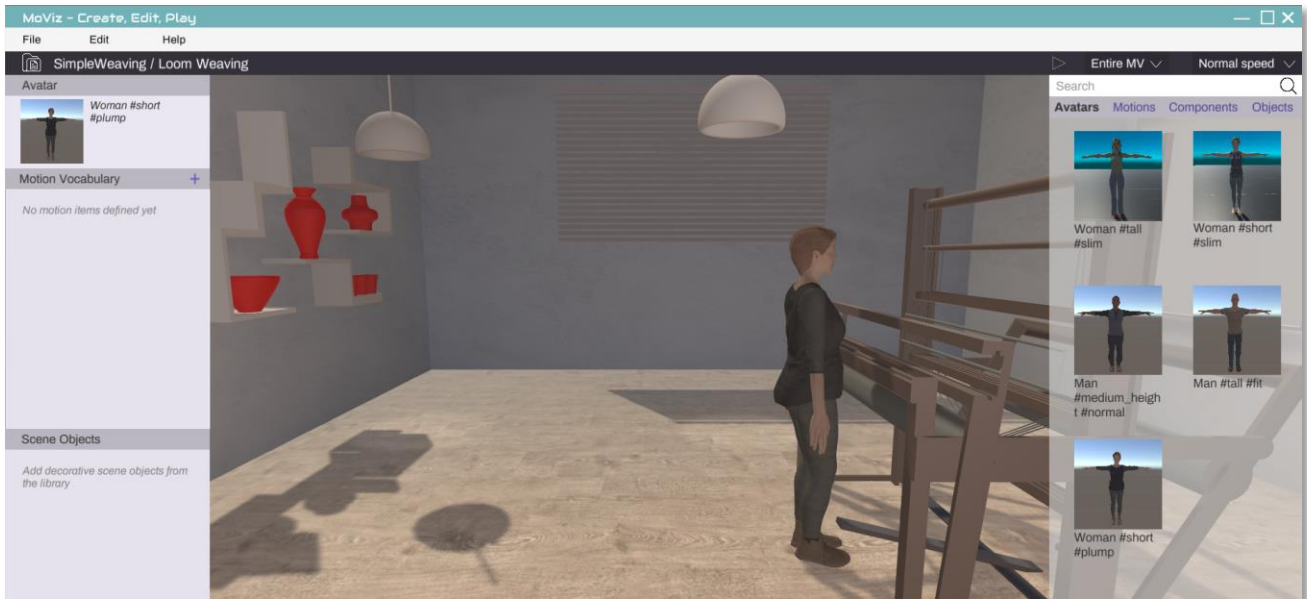


Figure 32: Design 3.0 - MSE - Avatar added to Scene. (source: Mingei, 2020) [113]

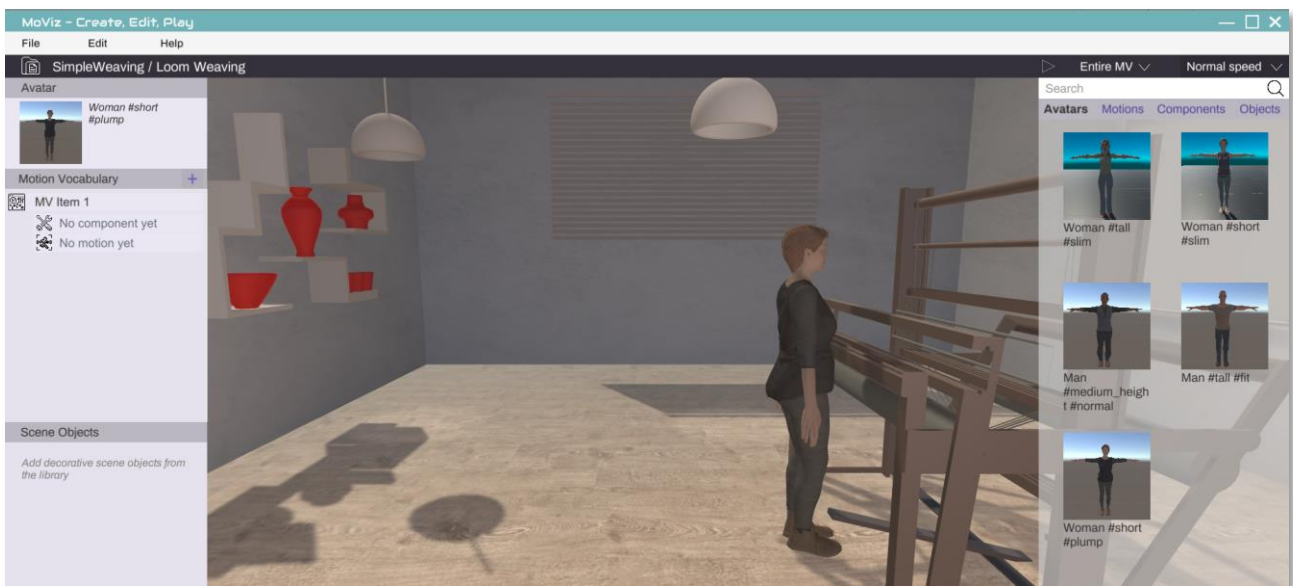


Figure 33: Design 3.0 - MSE - Motion Vocabulary Item added. (source: Mingei, 2020) [114]

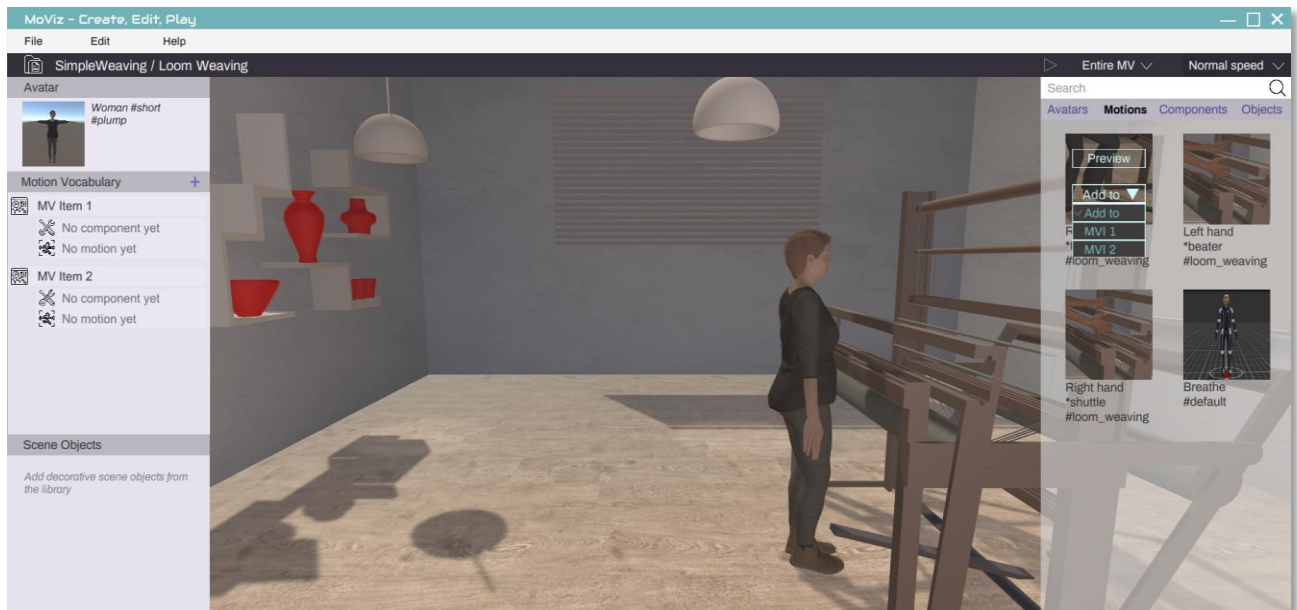


Figure 34: Design 3.0 - MSE - Second Motion Vocabulary Item added. (source: Mingei, 2020) [115]

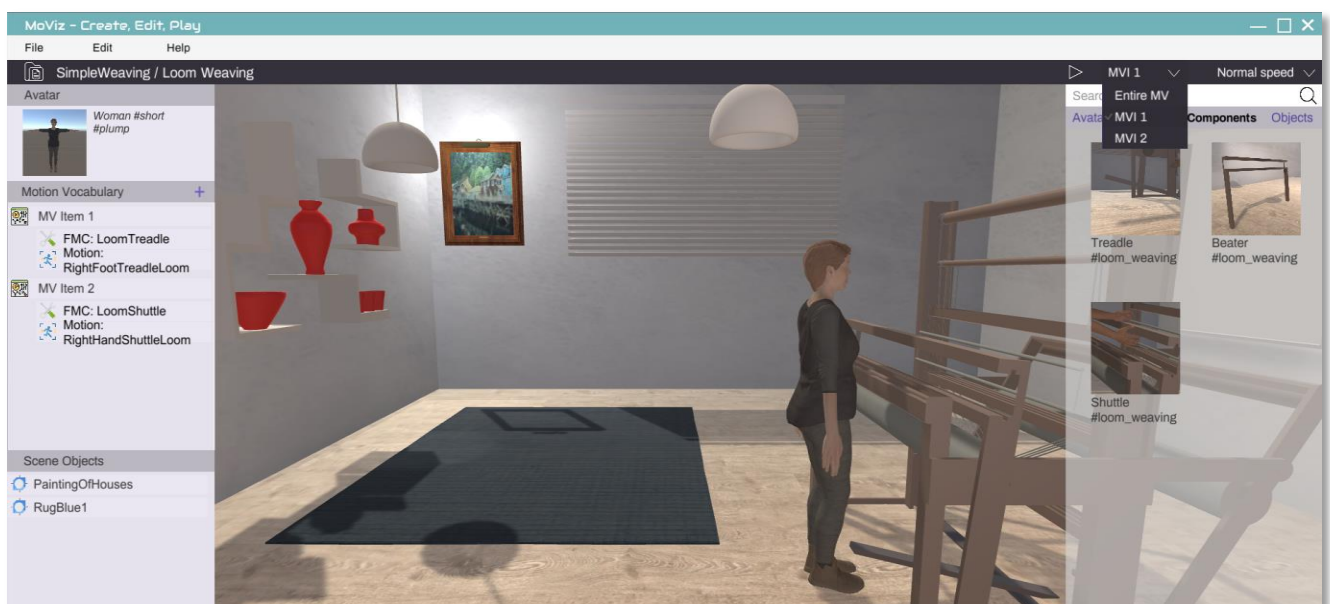


Figure 35: Design 3.0 - MSE - Motion Vocabulary Items are filled with Motions and FMCs, and Scene Objects have also been added. (source: Mingei, 2020) [116]

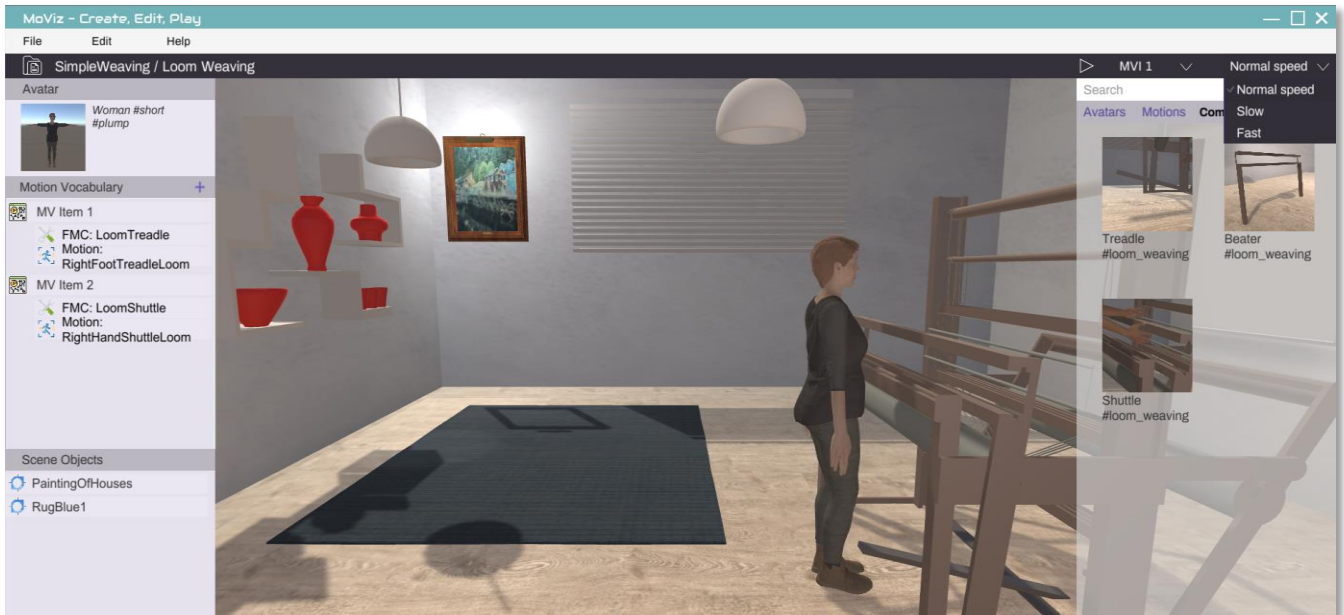


Figure 36: Design 3.0 - MSE - Preview of Animation Speed Selection Dropdown. (source: Mingei, 2020) [117]

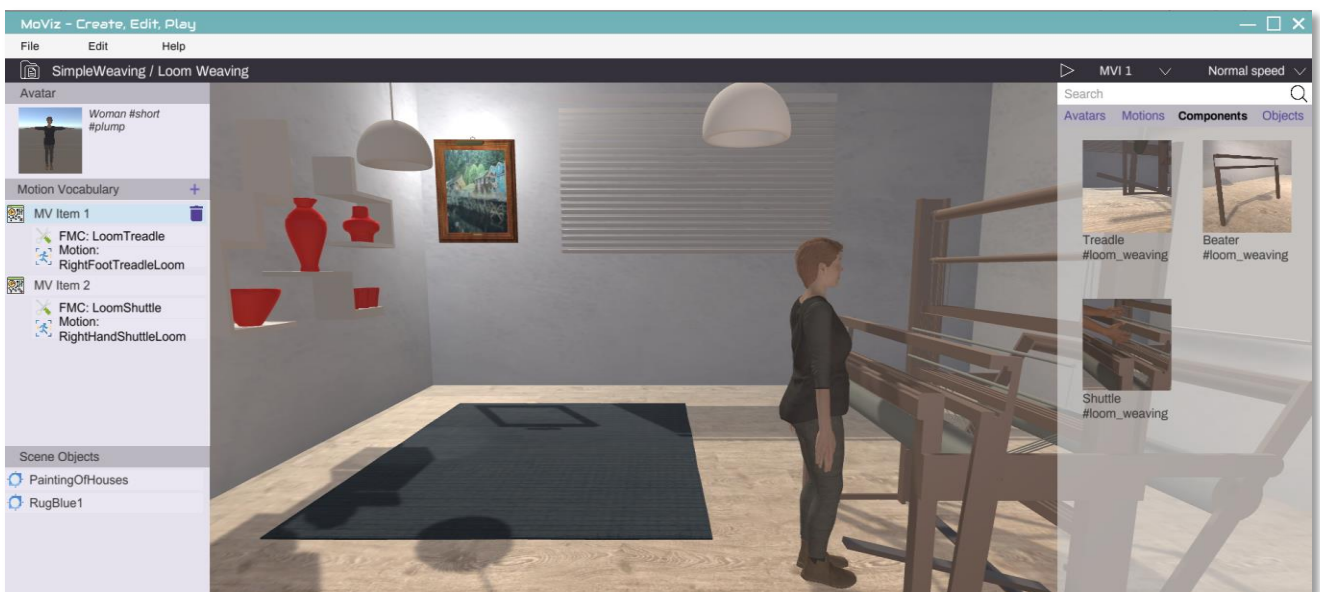


Figure 37: Design 3.0 - MSE - Preview of ability to delete an MVI by clicking on it. (Source: Mingei, 2020) [118]

6.3 Interacting with MoViz

Not only the design, but also the ways users can interact with the platform were iteratively refined according to the comments received during the Expert Based evaluations, as can also be seen by the screenshots of the system presented in the previous section. The way that users can interact with MoViz is described in the subsequent subsections.

6.3.1 Overview

The MoViz platform entails both the Motion and Scene Editor (MSE) and the Motion and Scene Player (MSP), presented to the user as a uniform Interface (see figures 30-41). Users can choose to create a new scene, open a pre-existing one, and play a scene either in 3D or VR mode. The scenes can consist of: (i) an Avatar (ii) the motion(s) they will execute, (iii) the tools and machines they will use during this motion and (iv) any objects the user wishes to add to the Scene (e.g. a rug on the floor). Each Scene regards a certain craft, so when creating a new Scene users need to select the craft they wish to explore in this scene. Moreover, depending on the selected craft, the respective machines and tools related to this craft are automatically added to the Scene.

6.3.2 Editing Mode

Users have a complete overview and control of what is happening in the Scene. On the top left side of the screen, they can see the Scene's Name and selected Craft. Directly below that is a section containing a picture of the Avatar they have selected, its description, as well as the Motion Vocabulary and Scene Objects they have added to the Scene. In the center of the screen is the "3D world" where users can see the selected Avatar, the machine of the craft that this scene regards, and all the Objects they add to the Scene. Each craft comes with a preselected "template" Scene, which includes any machines involved in it; for instance, in the case of loom weaving, the Scene will already contain a 3D model of the loom machine.

On the top right side of the screen is the Library, with all available Avatars, Motions, Components (FMCs) and Objects. Upon hovering on each of these items, users have the option to either Preview the item or add it to the Scene. In the case of Motions or Components, they also specify to which MVI they wish to add this item.

The first thing that should be done when creating a new Scene is the selection of a Virtual Human to add to the Scene (this is why the button to Add a new Motion Vocabulary Item is disabled until the user does so). However, users can still interact with the scene before selecting a VH: they can browse through the Library and can even add or delete Scene Objects. After selecting the Avatar they wish, users proceed to add one or more Motion Vocabulary Items. They can change the Avatar at any time, by selecting to add a different one from the Library. This is also true for any added Component or Motion; all users have to do to replace one is to select to add a new one from the Library. Users can click at any point in time on an item visible in the Overview (left side of the screen), and select to delete it, which removes it from both the Overview as well as the 3D Scene.

Regarding the playback while being in the Scene Editor, users have the ability to preview a single Motion Vocabulary Item, or the entire Motion Vocabulary, provided that each is fully completed, i.e. includes an assigned Component and an assigned Motion. The playback toolbar is visible on the upper right side of the screen, with two (2) dropdowns for selecting (i) which item they wish to play, and (ii) the speed of the playback (normal, slow or fast).

6.3.3 Playback Mode

After completing the authoring of the Scene, and even previewing the created MV, users can choose to export the Scene. The exported Scene can then be used as the input for the modules of

the platform that allow playing the entire Scene in 3D or VR. In VR mode, users can choose to activate the training mode, where an Avatar shows them how to execute a craft, step by step. In the simple VR mode, users are immersed in the world of the Virtual Human performing the craft, and can view from up close and from different angles all the movements of the Avatar, as well as the resulting movement of the machine parts and tools the Avatar is using. These options are available on MoViz's start screen. In all three (3) cases, before loading the simulation, users can rearrange the order in which the MVIs will be played, as well as their speed.

When selecting to perform VR training, users can only simulate the Motion Vocabulary Items that regard operations carried out by the hands, as the VR system's right and left hand controllers are used. During this simulation, users can see the hands of the Avatar in VR, showing them which tools they need to use and how to operate them to complete an action correctly. In this VR training, the ideal trajectories for the movements and the usage of the tools are visualized, and user trajectories are recorded, so as to compare them and provide the users with feedback regarding their performance (percentage of correctness, what they did wrong, in which step etc.).

In more detail, the FMCs are loaded as holograms, with indicative animations of how they should be moved and used, based on the original function of the machine/tool, so as to emulate the act of weaving in a simple manner. To further elaborate, these "holograms" are basically the same 3D representations of the machines and tools used for each movement, but they are explicitly textured differently than the original ones, so that they resemble a hologram version, acting as guidelines for the user to know which tool to use, as well as how to use it. Regarding the latter, there exist indicative arrows, showing which way a tool should be translated or rotated.

6.4 Playback in 3D

In the context of developing MoViz, the focus has mainly lied on the Motion and Scene Editor, which allows users to author craft experiences, since most of the platform's complexity, as well as the interaction of users with the platform's UI resides there. However, already in the editing mode of the platform (MSE), users can preview the Motion Vocabularies they create. As it is visible in the screenshots of the authoring tool in the section above, users can choose which MVI they wish to playback, as well as its speed. When exporting the scene, they can then play it in a 3D or VR Virtual Environment. This choice is available in the Start Screen of the platform (Figure 29). From there, they can configure the order of the MVIs, as well as their speed for the playback in 3D or VR. The User Interface for this part of the platform is currently under development, and constitutes part of our future work.

7 VR training

Regarding the VR training part, we have already created simple demos of using handheld tools, e.g., a hammer, as can be seen in Figure 42 below. It is important to note that for the VR training module, the ORamaVR SDK [87] is used. It is part of our immediate future work to progress further from handheld tools, to integrate FMCs in the VR training module, starting with the case of loom weaving, so as to have a complete representation of the craft in all the modules of MoViz.



Figure 38: Screenshots of the VR training module - showing to the user how to operate a hammer. (Source: Mingei, 2019) [119]. Online video at: <https://youtu.be/wpYxf-ZBFII>

Later on in the course of the first year of the project and after the Chios Meeting a second VR training prototype was implemented focusing on the craft of mastic cultivation.

8 Conclusions

PART A presented the motion visualization module that will be used for the representation of the craft as it was performed by the expert with the use of 3D avatars. The main purpose of was to develop a generic methodology for presenting craft experiences in Virtual Environments, by employing Virtual Humans as practitioners. The focus also lies on the importance of reenacting Heritage Crafts in particular, due to their importance and value, but at the same time the lack of a comprehensive methodology for their visualization in Virtual Environments. The approach put forward in this work proposes a visualization aimed to be attractive and engaging to the general public, aiming to help in preserving the Heritage Crafts through different dimensions:

1. Their promotion and dissemination, as the applications of the developed methodology could benefit various stakeholder groups, ranging from museum curators and content holders, to craft enthusiasts and the general public.
2. Their presentation and reenactment in a manner that is as accurate as possible, which entails not only the realistic use of the machines and tools involved in the craft by the Virtual Humans representing the craftspersons, but also the representation of both the tangible and intangible aspects of Heritage Crafts. To that end, craftspersons are centrally involved in the process of the conceptual decomposition of the crafts, which will allow their transfer to the digital world, as they need to provide functional insight and emic understanding of the represented process.

The methodology proposed consists of several steps, the development of which was the result of studying bibliography regarding crafts and tools, and was significantly facilitated by collaborative sessions with craft practitioners, using the pilot case of loom weaving as a template. These sessions took place in the context of the Mingei plenary meeting and co-creation sessions, at Haus der Seidenkultur in Krefeld Germany. The proposed methodology for transferring a craft from the physical to the digital world namely consists of the following steps:

1. Understanding the craft in question and performing conceptual decomposition of the craft
2. Identifying whether this craft includes (only) the use of handheld tools or (also) machines
3. In the case that the craft utilizes machines, performing a decomposition of the machines into simple machines, called Fundamental Machine Components
4. Performing Motion Capture of the practitioners
5. Associating the tools and machines used with their corresponding motions
6. Animating the Fundamental Machine Components by inducing their motion from the human motion

Subsequently, two (2) applications of the proposed methodology that were developed in its context, (i) for the case of Virtual Humans operating handheld tools, and (ii) for the case of Virtual Humans reenacting the heritage craft of loom weaving were discussed. For the first application, the ToolY platform is presented, which allows the visualization of VHs operating handheld tools (e.g., a hammer) in 3D or VR environments. For the second application, a comprehensive platform is developed, called MoViz, utilizing ToolY as a starting point, but providing users with an integrated solution for authoring their own scenes, reenacting craft experiences, which can include handheld tools but also machines. In the context of this Thesis, the MoViz case study has focused on loom weaving, while the approach used is generic, able to represent a multitude of crafts.

The contributions of this work can thus be summarized as follows: we deliver (i) a novel, comprehensive methodology for craft visualization in Virtual Environments, (ii) an authoring tool for craft experiences, which allows even non-technical users to utilize the results of complex technologies to author their own scenes, reenacting craft procedures, and (iii) a visualization of the authored scenes in 3D and VR environments, with the option of performing VR training for the handheld machine parts and tools. Finally, we claim that (iv) our approach for the presentation and reenacting of crafts in VEs can help in the representation and dissemination of Heritage Crafts, and thus contribute to the efforts of their preservation.

Challenges - Open Issues

Motion Retargeting comprises a known issue in the Computer Graphics Community, which, albeit well-studied, does not have one clear and simple solution [123] [124] [125]. In the scope of this Thesis, this affects the motion of the VHs, according to the Motion Capture animation files. In more detail, the way in which the VH is animated is by motion retargeting of the skeleton in the MoCap file, to the humanoid skeleton of the Avatar representing the VH. In many cases, depending on both the skeleton of the human that was recorded in the MoCap, as well as the build of the skeleton of the Avatar, the retargeting is not perfect. This means that the movement of the skeleton according to the MoCap might result in occlusions and “weird” behavior, e.g., a hand can move too close to the Avatar’s body and look as if it’s passing through it.

In the context of Mingei, currently, this issue is addressed using Unity’s Avatar Muscle & Settings¹³, for configuration of the degrees of freedom of joints in the skeleton of the Avatar. In more detail, for each of the available Virtual Humans the user can add to the scene, we have configured their range of motion, in order to ameliorate the deformation of the Avatar during the animations. However, in the future we would like to explore other possibilities for online motion retargeting, to minimize the manual configuration required and allow for a more automated process.

¹³ <https://docs.unity3d.com/Manual/MuscleDefinitions.html>

9 Future Work

Planned future work includes tackling all unaddressed issues discovered during both the expert and user based evaluations. A particularly important aspect of these issues that will be prioritized regards the improvement of the Avatars' movements for a more realistic representation, by investigating how to ameliorate the motion retargeting from the available MoCaps, as well as exploring different animation files to edit and select possibly better choices. Moreover, this will be facilitated by substituting the loom model used in the visualization with a 3D reconstructed one of the actual loom used during the MoCap sessions. Another significant issue that will be addressed regards the elimination of abbreviations and "technical terms" altogether, to increase the simplicity and user-friendliness of the application. Furthermore, we plan on working on the full implementation of MoViz, which includes work on the player mode (Motion and Scene Player - MSP), as well as the VR training.

Additionally, future work will include the addition of a Narrator in the scenes, i.e. the users will have the ability to add a second Virtual Human to their scenes, which will serve as a storyteller. This will require the careful consideration and redesign of the authoring tool, in order to include this capability in the platform seamlessly and in a user-friendly way. To that end, we plan to introduce a Storytelling Mode, which will provide users with the necessary editing tools (adding the Narrator, configuring its position in the Scene, as well as setting a timeline for its stories etc.), as well as the content (images, presentations, videos, text and audio files) for them to utilize.

Moreover, we plan to introduce a Craft Editor module to the MoViz platform. At the moment, the MoViz platform allows its users to author scenes regarding only certain predefined crafts, which are accompanied by a template scene, including the basic machines and tools involved in the craft in question. With the Craft Editor, we essentially plan to give the opportunity to users to create their own template scenes for crafts of their choice. To that end, the Craft Editor will need to offer functionality for importing and configuring the motions associated with the craft, as well as the machines and tools, and for the definition of the Fundamental Machine Components involved. Regarding the latter, the editor should offer some templates of basic FMCs that they can use, since most of them require joints for their correct articulation and movement.

In addition to creating their own Crafts, we would like to give users the possibility to add their own Avatars as well as Scene Objects. Although less complex than the craft template scenes creation, this also requires major work, since, besides the necessary additions to the platform to facilitate the addition of Avatars and other 3D objects, the User Interface should provide the users with appropriate configuration tools for the added items. In more details, spatial registration of the Avatars and Objects will need to take place, for example to scale them and translate them in the scene. Furthermore, a requirement for the Avatars is that they have to be rigged, i.e., include a humanoid skeleton. Additional configuration for the Avatars might also need to take place, for example with Unity's Avatar Muscle & Settings¹⁴, for configuration of the character's range of motion to ensure the character deforms in a convincing way, free from visual artifacts or self-overlaps.

¹⁴ <https://docs.unity3d.com/Manual/MuscleDefinitions.html>

PART A - 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. Papagiannakis, G., Lydatakis, N., Kateros, S., Georgiou, S., Zikas, P.: Transforming medical education and training with VR using MAGES. In: *SIGGRAPH Asia 2018 Posters*. p. 83. ACM (2018)
3. Create 3D models, characters | Download Adobe Fuse (Beta), https://www.adobe.com/gr_en/products/fuse.html
4. Hecker, C., Raabe, B., Enslow, R.W., DeWeese, J., Maynard, J., van Prooijen, K.: Real-time motion retargeting to highly varied user-created morphologies. In: *ACM Transactions on Graphics (TOG)*. p. 27. ACM (2008)
5. Dariush, B., Gienger, M., Arumbakkam, A., Goerick, C., Zhu, Y., Fujimura, K.: Online and markerless motion retargeting with kinematic constraints. In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 191–198. IEEE (2008)
6. Monzani, J.-S., Baerlocher, P., Boulic, R., Thalmann, D.: Using an intermediate skeleton and inverse kinematics for motion retargeting. In: *Computer Graphics Forum*. pp. 11–19. Wiley Online Library (2000)
7. Bill L. Counterbalance Loom. <https://3dwarehouse.sketchup.com/model/a4d5115a90e3f5534cf6cee9a1fdf035/Counterbalance-Loom>.
8. Anni Albers and Nicholas Fox Weber. 2017. *On Weaving: New Expanded Edition*. Princeton University Press.
9. K.-S. Huang, C.-F. Chang, Y.-Y. Hsu, and S.-N. Yang, “Key probe: a technique for animation keyframe extraction,” *Vis. Comput.*, vol. 21, no. 8–10, pp. 532–541, 2005.
10. M. Wang, S. Guo, M. Liao, D. He, J. Chang, and J. Zhang, “Action snapshot with single pose and viewpoint,” *Vis. Comput.*, vol. 35, no. 4, pp. 507–520, 2019.
11. C. Kirmizibayrak, J. Honorio, X. Jiang, R. Mark, and J. K. Hahn, “Digital analysis and visualization of swimming motion,” *Int. J. Virtual Real.*, vol. 10, no. 3, pp. 9–16, 2011.
12. G. Pingali, A. Opalach, Y. Jean, and I. Carlbom, “Visualization of sports using motion trajectories: providing insights into performance, style, and strategy,” in *Proceedings Visualization, 2001. VIS’01.*, 2001, pp. 75–544.
13. J. Assa, Y. Caspi, and D. Cohen-Or, “Action synopsis: pose selection and illustration,” *ACM Trans. Graph. TOG*, vol. 24, no. 3, pp. 667–676, 2005.
14. M. G. Choi, K. Yang, T. Igarashi, J. Mitani, and J. Lee, “Retrieval and visualization of human motion data via stick figures,” in *Computer Graphics Forum*, 2012, vol. 31, pp. 2057–2065.
15. H.-J. Lee, H. J. Shin, and J.-J. Choi, “Single image summarization of 3D animation using depth images,” *Comput. Animat. Virtual Worlds*, vol. 23, no. 3–4, pp. 417–424, 2012.
16. G. Papagiannakis, N. Lydatakis, S. Kateros, S. Georgiou, and P. Zikas, “Transforming medical education and training with VR using MAGES,” in *SIGGRAPH Asia 2018 Posters*, 2018, p. 83.
17. N. Pfeiffer-Leßmann and T. Pfeiffer, “ExProtoVAR: A Lightweight Tool for Experience-Focused Prototyping of Augmented Reality Applications Using Virtual Reality,” in *International Conference on Human-Computer Interaction*, 2018, pp. 311–318.
18. K. Kotis, “ARTIST-a reAl-time low-effoRt mulTi-entity Interaction System for creaTing reusable and optimized MR experiences,” *Res. Ideas Outcomes*, vol. 5, p. e36464, 2019.
19. P. Xiberta and I. Boada, “A new e-learning platform for radiology education (RadEd),” *Comput. Methods Programs Biomed.*, vol. 126, pp. 63–75, 2016.
20. D. D. Sumadio and D. R. A. Rambli, “Preliminary evaluation on user acceptance of the augmented reality use for education,” in *2010 second international conference on computer engineering and applications*, 2010, vol. 2, pp. 461–465.
21. A. Alsumait and Z. S. Al-Musawi, “Creative and innovative e-learning using interactive storytelling,” *Int. J. Pervasive Comput. Commun.*, vol. 9, no. 3, pp. 209–226, 2013.
22. S. Greenwald et al., “Technology and applications for collaborative learning in virtual reality,” 2017.

23. S. Bouchard *et al.*, “Virtual reality compared with in vivo exposure in the treatment of social anxiety disorder: a three-arm randomised controlled trial,” *Br. J. Psychiatry*, vol. 210, no. 4, pp. 276–283, 2017.
24. X. Pan and A. F. de C. Hamilton, “Why and how to use virtual reality to study human social interaction: The challenges of exploring a new research landscape,” *Br. J. Psychol.*, vol. 109, no. 3, pp. 395–417, 2018.
25. L. Freina and M. Ott, “A literature review on immersive virtual reality in education: state of the art and perspectives,” in *The International Scientific Conference eLearning and Software for Education*, 2015, vol. 1, p. 10.1007.
26. P. Gamito *et al.*, “Cognitive training on stroke patients via virtual reality-based serious games,” *Disabil. Rehabil.*, vol. 39, no. 4, pp. 385–388, 2017.
27. F. Ganier, C. Hoareau, and J. Tisseau, “Evaluation of procedural learning transfer from a virtual environment to a real situation: a case study on tank maintenance training,” *Ergonomics*, vol. 57, no. 6, pp. 828–843, 2014.
28. A. G. Greenwald, “Cognitive learning, cognitive response to persuasion, and attitude change,” *Psychol. Found. Attitudes*, vol. 1968, pp. 147–170, 1968.
29. A. G. Gallagher *et al.*, “Virtual reality simulation for the operating room: proficiency-based training as a paradigm shift in surgical skills training,” *Ann. Surg.*, vol. 241, no. 2, p. 364, 2005.
30. Y.-C. Huang and S. R. Han, “An immersive virtual reality museum via second life,” in *International Conference on Human-Computer Interaction*, 2014, pp. 579–584.
31. R. D. Webster, “Corrosion prevention and control training in an immersive virtual learning environment,” University of Alabama at Birmingham, Graduate School, 2014.
32. B. Chang, L. Sheldon, M. Si, and A. Hand, “Foreign language learning in immersive virtual environments,” in *The Engineering Reality of Virtual Reality 2012*, 2012, vol. 8289, p. 828902.
33. T. Bastiaens, L. Wood, and T. Reiners, “New landscapes and new eyes: The role of virtual world design for supply chain education,” *Ubiquitous Learn.*, vol. 6, no. 1, pp. 37–49, 2014.
34. F. P. Rahimian, T. Arciszewski, and J. S. Goulding, “Successful education for AEC professionals: case study of applying immersive game-like virtual reality interfaces,” *Vis. Eng.*, vol. 2, no. 1, p. 4, 2014.
35. A. Angulo and G. V. de Velasco, “Immersive simulation of architectural spatial experiences,” *Blucher Des. Proc.*, vol. 1, no. 7, pp. 495–499, 2014.
36. S. Braun and C. Slater, “Populating a 3D virtual learning environment for interpreting students with bilingual dialogues to support situated learning in an institutional context,” *Interpret. Transl. Train.*, vol. 8, no. 3, pp. 469–485, 2014.
37. P. Zikas *et al.*, “Mixed reality serious games and gamification for smart education,” in *European Conference on Games Based Learning*, 2016, p. 805.
38. M. Chollet, N. Chandrashekhar, A. Shapiro, L.-P. Morency, and S. Scherer, “Manipulating the perception of virtual audiences using crowdsourced behaviors,” in *International Conference on Intelligent Virtual Agents*, 2016, pp. 164–174.
39. J. Rickel and W. L. Johnson, “Animated agents for procedural training in virtual reality: Perception, cognition, and motor control,” *Appl. Artif. Intell.*, vol. 13, no. 4–5, pp. 343–382, 1999.
40. D. Traum and J. Rickel, “Embodied agents for multi-party dialogue in immersive virtual worlds,” in *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2*, 2002, pp. 766–773.
41. Z. Paul, P. Margarita, M. Vasilis, and P. George, “Life-sized Group and Crowd simulation in Mobile AR,” in *Proceedings of the 29th International Conference on Computer Animation and Social Agents*, 2016, pp. 79–82.
42. L. Chittaro, R. Ranon, and L. Ieronutti, “Guiding visitors of Web3D worlds through automatically generated tours,” in *Proceedings of the eighth international conference on 3D Web technology*, 2003, pp. 27–38.
43. D. Economou, W. L. Mitchell, and T. Boyle, “Requirements elicitation for virtual actors in collaborative learning environments,” *Comput. Educ.*, vol. 34, no. 3–4, pp. 225–239, 2000.
44. R. Hertz-Lazarowitz and H. Shachar, “Teachers’ verbal behaviour in cooperative and whole-class instruction,” *Coop. Learn. Theory Res.*, pp. 77–94, 1990.

45. A. Hartholt *et al.*, “All together now,” in *International Workshop on Intelligent Virtual Agents*, 2013, pp. 368–381.
46. “Virtual Human Toolkit.” <https://vhtoolkit.ict.usc.edu/> (accessed Apr. 09, 2019).
47. J. Cassell, J. Sullivan, E. Churchill, and S. Prevost, *Embodied Conversational Agents*. MIT Press, 2000.
48. S. Baldassarri, E. Cerezo, and F. J. Seron, “Maxine: A platform for embodied animated agents,” *Comput. Graph.*, vol. 32, no. 4, pp. 430–437, Aug. 2008, doi: 10.1016/j.cag.2008.04.006.
49. W. Swartout *et al.*, “Virtual museum guides demonstration,” in *2010 IEEE Spoken Language Technology Workshop*, 2010, pp. 163–164.
50. J. C. Campbell, M. J. Hays, M. Core, M. Birch, M. Bosack, and R. E. Clark, “Interpersonal and leadership skills: using virtual humans to teach new officers,” in *Proc. of Interservice/Industry Training, Simulation, and Education Conference, Paper*, 2011, vol. 11358.
51. D. DeVault *et al.*, “SimSensei Kiosk: A virtual human interviewer for healthcare decision support,” in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, 2014, pp. 1061–1068.
52. R. Aylett, M. Vala, P. Sequeira, and A. Paiva, “Fearnot!—an emergent narrative approach to virtual dramas for anti-bullying education,” in *International Conference on Virtual Storytelling*, 2007, pp. 202–205.
53. J. Lee, J. Chai, P. S. Reitsma, J. K. Hodgins, and N. S. Pollard, “Interactive control of avatars animated with human motion data,” in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 491–500.
54. M. Mori, “Bukimi no tani [the uncanny valley],” *Energy*, vol. 7, pp. 33–35, 1970.
55. M. Seymour, K. Riemer, and J. Kay, “Interactive Realistic Digital Avatars-Revisiting the Uncanny Valley,” 2017.
56. B. Yao and L. Fei-Fei, “Modeling mutual context of object and human pose in human-object interaction activities,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 17–24.
57. G. Yu, Z. Liu, and J. Yuan, “Discriminative orderlet mining for real-time recognition of human-object interaction,” in *Asian Conference on Computer Vision*, 2014, pp. 50–65.
58. M. Kallmann and D. Thalmann, “Modeling objects for interaction tasks,” in *Computer Animation and Simulation’98*, Springer, 1999, pp. 73–86.
59. T. Abaci, J. Ciger, and D. Thalmann, “Planning with smart objects,” in *International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2005, no. ARTICLE.
60. L. Levison, “Connecting planning and acting via object-specific reasoning,” *Citeseer*, 1996.
61. H. B. Helbig, J. Steinwender, M. Graf, and M. Kiefer, “Action observation can prime visual object recognition,” *Exp. Brain Res.*, vol. 200, no. 3–4, pp. 251–258, 2010.
62. V. Kaptelinin and B. Nardi, “Affordances in HCI: toward a mediated action perspective,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, pp. 967–976.
63. J. J. Gibson, *The theory of affordances*. R. Shaw and J. Bransford (eds.), *Perceiving, Acting and Knowing*. Hillsdale, NJ: Erlbaum, 1977.
64. J. J. Gibson, *The ecological approach to visual perception*. Boston, MA, US: Houghton, Mifflin and Company, 1979.
65. D. A. Norman, *The psychology of everyday things*. Basic books, 1988.
66. M. Hassan and A. Dharmaratne, “Attribute based affordance detection from human-object interaction images,” in *Image and Video Technology*, 2015, pp. 220–232.
67. H. Kjellström, J. Romero, and D. Kragić, “Visual object-action recognition: Inferring object affordances from human demonstration,” *Comput. Vis. Image Underst.*, vol. 115, no. 1, pp. 81–90, 2011.
68. B. Yao, J. Ma, and L. Fei-Fei, “Discovering object functionality,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2512–2519.
69. “Simple machine,” *Wikipedia*. Mar. 04, 2020, Accessed: Mar. 04, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Simple_machine&oldid=943804888.
70. P. Zikas, N. Lydatakis, S. Kateros, and G. Papagiannakis, “Scenior: An Immersive Visual Scripting system of Gamified Training based on VR Software Design Patterns,” *ArXiv Prepr. ArXiv190905719*, 2019.

71. M. Patkin, “A Check-List for Handle Design,” p. 22.
72. A. R. PAUL, P. Roy, and S. MUKHERJEE, *Mechanical sciences: Engineering mechanics and Strength of Materials*. PHI Learning Pvt. Ltd., 2004.
73. A. P. Usher, *A History of Mechanical Inventions*. Courier Corporation, 1954.
74. I. Asimov, *Understanding Physics*. Dorset Press, 1988.
75. G. D.-I. 200 (2002 Rostock, Interactive Systems. Design, Specification, and Verification: 9th International Workshop, DSV-IS 2002, Rostock Germany, June 12-14, 2002. Springer, 2002.
76. W. B. Anderson, *Physics for Technical Students: Mechanics and heat*. 1st ed. McGraw-Hill book Company, 1914.
77. <https://www.britannica.com/technology/simple-machine>
78. E. L. Prater, *Basic Machines*. Echo Point Books & Media, 1994.
79. B. of N. Personnel, *Basic Machines and How They Work*. Dover Publications, 1971.
80. A. Albers and N. F. Weber, *On Weaving: New Expanded Edition*. Princeton University Press, 2017.
81. N. Kantola and T. Jokela, “Svsb: simple and visual storyboards: developing a visualization method for depicting user scenarios,” in *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*, 2007, pp. 49–56.
82. E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, and N. M. Thalmann, “ToolTY: An approach for the combination of motion capture and 3D reconstruction to present tool usage in 3D environments,” in *Intelligent Scene Modelling and Human Computer Interaction*, N. M. Thalmann, J. Zhang, and J. Zheng, Eds. Springer, 2020.
83. B. Dariush, M. Gienger, A. Arumbakkam, C. Goerick, Y. Zhu, and K. Fujimura, “Online and markerless motion retargeting with kinematic constraints,” in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 191–198.
84. C. Hecker, B. Raabe, R. W. Enslow, J. DeWeese, J. Maynard, and K. van Prooijen, “Real-time motion retargeting to highly varied user-created morphologies,” in *ACM Transactions on Graphics (TOG)*, 2008, vol. 27, p. 27.
85. J.-S. Monzani, P. Baerlocher, R. Boulic, and D. Thalmann, “Using an intermediate skeleton and inverse kinematics for motion retargeting,” in *Computer Graphics Forum*, 2000, vol. 19, pp. 11–19.
86. E. Stefanidi, N. Partarakis, X. Zabulis, and G. Papagiannakis, “An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments,” in *International Conference on Human-Computer Interaction*, to appear 2020.
87. “ORamaVR – from the operating room to VR.” <http://oramavr.com/> (accessed Feb. 24, 2020).
88. Comparison of incorrect and correct postures for 3 handheld tools: a seesaw, pliers and a hammer.
89. E. Stefanidi, N. Partarakis, X. Zabulis, The proposed methodology for Craft Visualization, (2019) [Diagram]. Retrieved through screen capture, Bibliographical reference: Unpublished research work
90. Source: LinkedIn, (2019) The parts that a loom machine consists of. [Illustration]. Retrieved from <https://www.linkedin.com/pulse/20141030195455-49457671-understanding-weaving-what-are-ooms/>
91. Source: Mingei, (2019) Co-design session at Haus der Seidenkultur (HdS), Krefeld, Germany , [Picture]. Retrieved during the co-creation session
92. Source: Mingei, (2019) Motion Capture sessions of a practitioner while loom weaving at HdS , Krefeld. [Picture compilation]. Retrieved during MoCap sessions
93. Images of loom components retrieved from Wikipedia, Basic loom components. (2019) [Pictures compilation] Retrived by Wikipedia, compiled and edited by Mingei
94. Adami, I., Karuzaki, E., N. Partarakis, X. Zabulis, Storyboard of the three stages of weaving and the machine parts involved. (2019) [Illustration/screenshot]. Created with and retrieved from PowerPoint
95. Adami, I., Karuzaki, E., N. Partarakis, X. Zabulis, Overview of the weaving process: steps, actions, and FMCs involved. (2019) [Illustration/screenshot]. Created with and retrieved from PowerPoint
96. E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, Overview of ToolTY’s pipeline, (2020) [Screen Capture]. Retrieved through screen capture from ToolTY, Bibliographical reference: E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, “ToolTY: An approach for the combination of motion capture and 3D reconstruction to present tool usage in 3D

- environments", In Intelligent Scene Modelling and Human Computer Interaction, Thalmann N., Zhang J, Jiang X. (eds.), Springer, 2020.
97. E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, Screenshot of the 2 Virtual Humans holding (a) a hammer and (b) scissors, (2020) [Screen Capture]. Retrieved through screen capture from ToolTy, Bibliographical reference: E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, "ToolTY: An approach for the combination of motion capture and 3D reconstruction to present tool usage in 3D environments", In Intelligent Scene Modelling and Human Computer Interaction, Thalmann N., Zhang J, Jiang X. (eds.), Springer, 2020.
 98. E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, Screenshots of a Virtual Human holding and operating a hammer, (2020) [Screen capture compilation]. Retrieved through screen capture from ToolTy, Bibliographical reference: E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, "ToolTY: An approach for the combination of motion capture and 3D reconstruction to present tool usage in 3D environments", In Intelligent Scene Modelling and Human Computer Interaction, Thalmann N., Zhang J, Jiang X. (eds.), Springer, 2020.
 99. E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, Grip points, orientations and resulting attachment of hammer to the VH hand r, (2020) [Screen capture compilation]. Retrieved through screen capture from ToolTy, Bibliographical reference: E. Stefanidi, N. Partarakis, X. Zabulis, P. Zikas, G. Papagiannakis, N. Thalmann, "ToolTY: An approach for the combination of motion capture and 3D reconstruction to present tool usage in 3D environments", In Intelligent Scene Modelling and Human Computer Interaction, Thalmann N., Zhang J, Jiang X. (eds.), Springer, 2020.
 100. E. Stefanidi, N. Partarakis, X. Zabulis, Loom treadle model in its "max" and "min" positions, with joints visible, (2019) [Screenshot]. Retrieved through screen capture in Maya, Bibliographical reference: Unpublished research work
 101. E. Stefanidi, N. Partarakis, X. Zabulis, Loom beater model in its idle state and "max", "min" positions, with joints visible, (2019) [Screenshot]. Retrieved through screen capture in Maya, Bibliographical reference: Unpublished research work
 102. E. Stefanidi, N. Partarakis, X. Zabulis, Downloaded loom model VS. model after editing (noticeable difference in the treadle mechanism)., (2019) [Screenshot compilation]. Retrieved through screen capture in Maya, Bibliographical reference: Unpublished research work
 103. Stefanidi, E., Partarakis, N., Zabulis, X. (2019) Visualization of the foot pressing the treadle. [Screen capture]. Retrieved through screen capture from Unity3D, Bibliographic Reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
 104. Stefanidi, E., Partarakis, N., Zabulis, X. (2019) Attaching the hands on the beater. [Technical Drawing]. Retrieved through screen capture from Unity3D, Bibliographic Reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
 105. Stefanidi, E., Partarakis, N., Zabulis, X. (2019) Attaching the hands on the shuttle. [Technical Drawing]. Retrieved through screen capture from Unity3D, Bibliographic Reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
 106. Stefanidi, E., Partarakis, N., Zabulis, X. (2019) Visualization of the result: the VH is operating the loom. [Screen capture]. Retrieved through screen capture from Unity3D, Bibliographic Reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
 107. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MoViz Start Screen., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work

108. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MoViz Start Screen - Add New Scene Selected., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
109. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MoViz Start Screen - Open Scene Selected., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
110. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Empty Scene., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
111. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - empty Scene, hovering over Avatar in Library., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
112. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Avatar Preview., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
113. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Avatar added to Scene., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
114. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Motion Vocabulary Item added., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
115. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Second Motion Vocabulary Item added., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
116. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Motion Vocabulary Items are filled with Motions and FMCs, and Scene Objects have also been added., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
117. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Preview of Animation Speed Selection Dropdown. , (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
118. E. Stefanidi, N. Partarakis, X. Zabulis, Design 3.0 - MSE - Preview of ability to delete an MVI by clicking on it., (2020) [Screenshot] Retrieved through screen capture of MoViz, Bibliographical reference: Unpublished research work
119. E. Stefanidi, N. Patsiouras, Zikas, P., N. Partarakis, Papagiannakis, G., X. Zabulis, Screenshots of the VR training module - showing to the user how to operate a hammer [Screenshots compilation]. Retrieved through screen capture of the VR camera in Unity3D, Bibliographical reference: Unpublished research work
120. E. Stefanidi, N. Partarakis, X. Zabulis, Decomposition of loom weaving into steps. (2019) [Table], Bibliographical reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
121. E. Stefanidi, N. Partarakis, X. Zabulis, Main machine parts involved in loom weaving. (2019) [Table], Bibliographical reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
122. E. Stefanidi, N. Partarakis, X. Zabulis, Materials and products identified for loom weaving. (2019) [Table], Bibliographical reference: Stefanidi, E., Partarakis, N., Zabulis, X., Papagiannakis, G. (2020), An Approach for the Visualization of Crafts and Machine Usage in Virtual Environments to appear in the proceedings of ACHI 2020 (Thirteenth International Conference on Advances in Computer-Human Interactions)
123. B. Dariush, M. Gienger, A. Arumbakkam, C. Goerick, Y. Zhu, and K. Fujimura, "Online and markerless motion retargeting with kinematic constraints," in 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008, pp. 191–198.

124. C. Hecker, B. Raabe, R. W. Enslow, J. DeWeese, J. Maynard, and K. van Prooijen, “Real-time motion retargeting to highly varied user-created morphologies,” in *ACM Transactions on Graphics (TOG)*, 2008, vol. 27, p. 27.
125. J.-S. Monzani, P. Baerlocher, R. Boulic, and D. Thalmann, “Using an intermediate skeleton and inverse kinematics for motion retargeting,” in *Computer Graphics Forum*, 2000, vol. 19, pp. 11–19.

PART B – Detailed Table of Contents

1. Introduction	73
2. State of the art of pose estimation frameworks and gesture recognition methods	74
2.1. Capturing RGD-B frames	74
2.1.1. iphone XS camera	74
2.1.2. Intel RealSense Depth Camera D435	75
2.1.3. Microsoft's Xbox Kinect	76
2.2. Pose estimation	76
2.2.1. Pose estimation frameworks	76
2.2.2. OpenPose	77
2.3. Gesture recognition methods	79
2.3.1. Dynamic Time Warping (DTW)	79
2.3.2. Hidden Markov Models (HMMs)	79
2.4. Evaluation framework and metrics	80
3. Data recording for Pose Estimation	82
3.1. Data recording using RGB-D cameras	82
3.2. Pose estimation: 3d extraction from 2D provided by Open Pose	83
3.3. Gesture recognition	83
3.4. The datasets used for pose estimation and gesture recognition	84
3.4.1. The Glass Blowing dataset	84
3.4.2. The Silk Weaving dataset	86
3.4.3. The Mastic dataset	87
4. Evaluation	89
4.1. Comparative evaluation	89

4.1.1.	Pose estimation challenges	89
4.1.2.	Comparison of pose estimation results in 2 datasets	92
4.1.3.	Gesture recognition with data from Silk Weaving dataset	92
4.1.4.	Gesture recognition comparisons using the Glass blowing dataset: 2d vs 3d, 2 joints vs 7 joints	93
4.2.	Conclusion	96
5.	Conclusions and Future Work	97
PART B - References		98

PART B - List of figures

Figure 1: iPhone XS back camera (Dual Rear Camera) (source: ICVS conference, 2019) [2]	75
Figure 2: iPhone XS front camera (True Depth Camera) capturing depth (source: ICVS conference, 2019) [2].....	75
Figure 3: Architecture of Intel RealSense Depth Camera D435 (source: Master Thesis, UAM Polytechnic School, 2019) [32]	76
Figure 4: OpenPose pipeline Reprinted from [4] (source: IEEE MultiMedia, vol. 19, 2012)	78
Figure 5: Illustration of the 2 cameras used (source: www.intel.com, www.apple.com) [23], [24]	82
Figure 6: Gesture recognition pipeline followed by the application (source: Autonomous Robots) [21]	84
Figure 7: Example of skeleton estimation in Silk Weaving and Glass Blowing dataset (source: Mingei, 2019) [25], [26]	90



PART B - List of tables

Table 1: List of popular open-source frameworks (source: ICVS conference, 2019) [2]	79
Table 2: Example of confusion matrix (source: Master Thesis, UAM Polytechnic School, 2019) [32]	80
Table 3: Example of frames from Glass Blowing dataset (source: compiled by Mingei, 2019) [25]	85
Table 4: Example of frames recorded with different camera's angles in the Glass blowing dataset (source: compiled by Mingei, 2019) [25]	86
Table 5: Example of frames from the Silk Weaving dataset (source: ICVS conference, 2019) [2]	87
Table 6: Example of frames from the Mastic cultivation dataset (source: compiled by Mingei, 2019) [27]	88
Table 7: Examples of skeleton estimation in the mastic dataset (source: compiled by Mingei, 2019) [26]	90
Table 8: Example of pose estimation in images recorded with different camera's angles in the Glass blowing dataset (source: compiled by Mingei, 2019) [25]	92
Table 9: OpenPose results on the Silk Weaving and Glass Blowing datasets (source: Master Thesis, UAM Polytechnic School, 2019) [32]	92
Table 10: Gesture recognition results using 2 joints and 2 dimensions on Silk Weaving dataset (source: ICVS conference, 2019) [2]	93
Table 11: Gesture recognition results comparison between 2D versus 3D for 2 joints in the glass blowing dataset (source: Master Thesis, UAM Polytechnic School, 2019) [32]	94
Table 12: Gesture recognition results when using 2D for 2 joints and 4 classes instead of 6 (source: Diploma thesis, AUTH University, 2020) [31]	95
Table 13: Gesture recognition results when using 2D for 7 joints (source: Diploma thesis, AUTH University, 2020) [31]	95
Table 14: Mean f-score and total accuracy for the glassblowing dataset with 2 joints and 7 joints (source: Diploma thesis, AUTH University, 2020) [31]	95



1. Introduction

Beyond preservation, dissemination and valorisation of Heritage Crafts MINGEI will make use of motion capture technologies to propose to the visitor of a museum to live a unique experience of imitating the movement of craftsman and receiving a real time feedback base on how well they performed. After having watched the 3D avatar of the expert performing the gestures the visitor will be invited to experience him or herself. In order to provide him/her a sonification feedback , visitor's body must be tracked first by a camera, and the position of his/her articulations must be detected. As explained in Part A, the Intangible Cultural Heritage that is being recorded in the previous task, will be demonstrated to students/tourists/visitors of cultural institutions such as museums through an interactive installation. Expert gestural data will be visualized in a 3D environment and integrated in a storytelling process, enriched with additional elements about the craft in general. Visitors will be invited to reproduce the gestures/actions virtually, to manipulate basic tools and their gestural performance will be tracked in real time with smart devices and cameras and low-cost motion capture sensors.

One of WP5 goal, described in this Part B, is to develop the tools and software for body tracking and gesture recognition. These tools will be used to develop the interactive mechanism that will be integrated in the installation permitting to simulate expert gestures and receive a feedback on how well the gestures have been performed by the museum visitor (Task and deliverable 5.3). More precisely in this task a deep learning based framework for human pose estimation has been used to track human body in real time and to provide accurate features that represent the position in 2 or 3 dimensional space. Video recordings of the three pilots have been done together with several tests in order to identify the most valuable features to achieve the best accuracy in gesture recognition. These experiments, together with the results are presented in this Part B of the deliverable.

2. State of the art of pose estimation frameworks and gesture recognition methods

The role of body tracking of professional actions, activities and gestures is of high importance for this task. Motion sensing and machine learning have actively contributed to the capturing of gestures and the recognition of meaningful movement patterns by machines. For this purpose, very interesting applications have emerged according to the industry needs. For example, machines that can continuously track and recognize human body and whereas in the creative and cultural industries it still remains a challenge to recognize and identify the motor skills of a given expert, will augment the capabilities of workers. Therefore, the accurate estimation of human body's pose is an important challenge. New cameras, even integrated in smartphones, are equipped with depth sensors and high-power processors, which allow us to record data even without very sophisticated devices.

2.1. Capturing RGD-B frames

The current market offers a wide variety of devices capable of capturing RGB and depth frames (RGB-D frames), however, in this task, we have focused on three different devices: iPhone XS True Depth camera and Dual Rear camera, Intel RealSense Depth Camera D435 and Microsoft's Xbox Kinect camera.

2.1.1. iphone XS camera

Since the launch of iPhone X in 2017, the new Apple phones have a depth mapping system in their rear and front camera.

On one hand, iPhone XS uses a dual rear camera, composed by a wide-angle lens and a telephoto lens capturing data with the same frame rate, to obtain the disparity, defined as the displacement in the position for corresponding points between the images. The main feature of the camera is that it is stereo rectified, which means that both cameras are pointing in the same direction and have the same focal length, distance from the focal point to the image plane and, in addition, the distance between the two optical lenses refers to the baseline.

In the **Figure 1(a)** is shown how the rear camera captures the normalized disparity, which mathematically equals to calculate the inverse of the depth (1/meters). First, the rays of lights from the observed object pass through the optical centres and are reflected on the image plane of each camera. Then, the mathematical relation tying the depth (Z), baseline distance (B), disparity (D) and focal length (F), showed in equation 2.1, results in the normalized disparity. Finally, the iPhone needs to filter and post-process the disparity to smooth the edges and fill the holes, which requires a heavy computation [1].

$$\frac{B}{Z} = \frac{D}{F} \rightarrow \frac{1}{Z} = \frac{D}{BF} \quad (2.1)$$

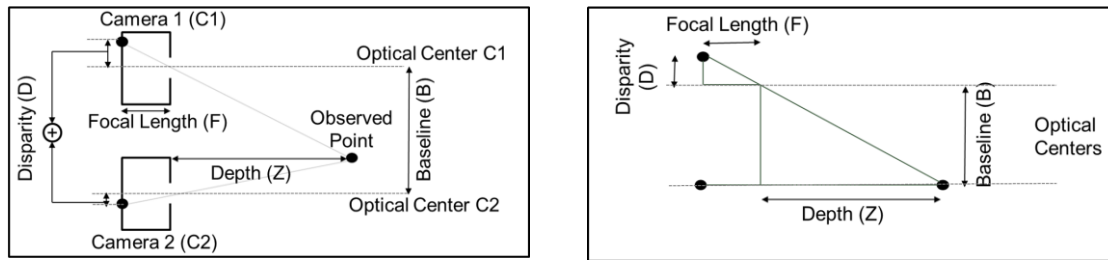


Figure 1: iPhone XS back camera (Dual Rear Camera) (source: ICVS conference, 2019) [2]

On the other hand, the front camera, named True Depthera by Apple, is used to measure the depth in meters directly. It has a dot projector that launches over 30,000 dots onto the scene, generally the user face, which are then captured by an infrared camera. To ensure that the system works properly in the dark, there is an ambient light sensor and a flood illuminator which adds more infrared light when needed. The final result is more stable depth images with a higher resolution, 640x480 instead of 320x240 obtained by the dual rear camera. **Figure 1(b)** shows the architecture of the True Depth Camera.

Finally, both cameras capture RGB-D frames with a frame rate of 30 fps and by using a portrait mode, which means subtracting the foreground, object that is usually a person who is focused, from the background.

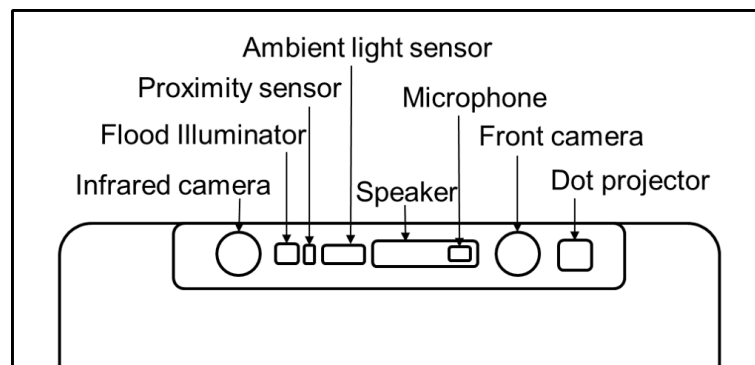


Figure 2: iPhone XS front camera (True Depth Camera) capturing depth (source: ICVS conference, 2019) [2]

2.1.2. Intel RealSense Depth Camera D435

The Intel RealSense Depth Camera D435 [3] is an USB-powered camera that includes depth sensors and a RGB sensor. It uses stereo vision to calculate depth and its implementation is based on a left imager, right imager, that capture depth in a similar way to the dual rear camera of the iPhone, and an optional infrared projector to improve depth accuracy by projecting a static infrared pattern on the scene to increase texture on low texture scenes. Moreover, it has a RGB module (Colour camera) to colour frames providing texture data, with whom we can make an overlaying on the depth to create a colour point cloud and a 3D reconstruction. The resolution is up to 1280 x 720 and the depth stream output frame rate is up to 90 fps.

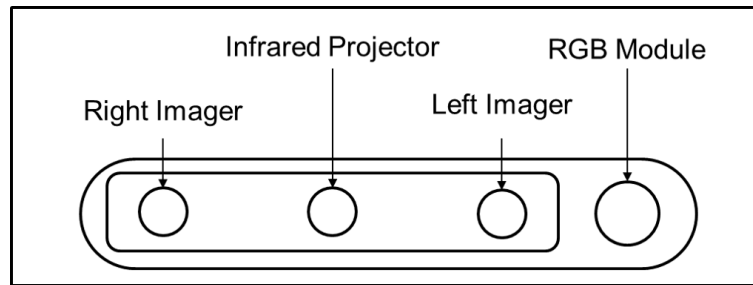


Figure 3: Architecture of Intel RealSense Depth Camera D435 (source: Master Thesis, UAM Polytechnic School, 2019) [32]

2.1.3. Microsoft's Xbox Kinect

Kinect [4] is a Microsoft motion sensor add-on for the Xbox 360 gaming console. It is composed of three main hardware pieces: a Colour VGA video camera, a depth sensor, and a multi-array microphone.

The VGA camera captures the RGB colour space as well as body-type and facial features. Its resolution is 640x480 pixels and has a frame rate of 30 fps. The depth sensor contains a monochrome CMOS sensor and an infrared projector that measures the distance of each point of the body of the player by transmitting invisible near-infrared light and measuring its "time of flight" after it reflects off the objects. Finally, it has a multi-array microphone that can isolate the voices of the player from other background noises.

2.2. Pose estimation

Due to the growing need to understand and predict human behaviour and motor skills, challenges such as human pose estimation become crucial in the field of Machine Learning. Human pose estimation can be defined as the problem of locating and representing in a coordinate system, point cloud, the set of keypoints or joints that shape a person into an image or video.

Two different approaches exist for pose estimation in single or multi-person scenarios: the top-down and the bottom-up approach. In the first approach, a human detector is initiated and both, the joints and the skeleton of each person, are calculated separately. This approach makes use of existing techniques for single-person pose estimation. However, the top-down approach suffers from an early commitment when the detector fails, and the computational power increases exponentially with the number of people in the scene.

In contrast, the bottom-up approach firstly detects and labels all the joint candidates in the frame. It secondly associates them to each individual person without using any person detector. It is, generally, a more complex approach, but is more robust to occlusion and complex poses.

2.2.1. Pose estimation frameworks

Different benchmarks of 2D human pose estimation are evaluated through annual challenges that aim to improve various key-parameters, such as the accuracy, how the algorithm performs with partial occlusions or using different keypoints, how to detect the pose of a big number of

individuals in the scene, etc. Some popular examples are the COCO Keypoint, MPII HumanPose and Posetrack challenges. Below, we will review and compare some of the popular top-down and bottom-up approaches for pose estimation.

2.2.1.1. AlphaPose

AlphaPose [5] is a top-down method based on regional multi-person pose estimation (RMPE). It follows a pipeline in which it is first applied the object detector Faster- RCNN[6] to obtain the human bounding boxes. This bounding box will fed into a Spatial Transformer Network (STN), which select the region of interest, then, a parallel single-person pose estimator (SPPE) module, to improve robustness against imperfect human bounding boxes, and finally, a Spatial Detransformer Network (SDTN), which generates pose proposals. At the end, a Parametric Pose Non-Maximum-Suppression (NMS) is carried out to eliminate redundant pose estimations

2.2.1.2. DensePose

DensePose [7], developed by Facebook AI research, takes as input an RGB image and estimates the surface-based description of the human body. This framework starts by applying an adapt version of Mask-RCNN with a Feature Pyramid Network (FPN)[8] features, FPN brings robustness to CNN in detection at different scales, to predict the discrete part labels and continuous surface coordinates. The reconstruction of the surfaces is carried out by an inpainting process based on another convolutional neural network, this process allows to recover deteriorated part of the image or body parts that are hidden and is relies on the estimation performed at different scales by the FPN.

2.2.1.3. Human Mesh Recovery

Human Mesh Recovery (HMR) [9] framework aims to map each person and extract a 3D joint surface from the shape and the angles of the human body by only using RGB images. The framework takes an input image cantered on a human in a feed-forward manner. Then, a convolutional encoder, bases on Skinned Multi-Person Linear (SMPL) [10] model, generates the features that will feed an iterative 3D regression module whose objective is to infer the 3D reconstruction of the human body. After that, HMR uses a discriminator network which selects the meshes that belong to the real human.

2.2.1.4. Deep cut

DeepCut [11] is an example of bottom-up approach, it uses an adapt Fast R-CNN version called AFR-CNN to extract a set of joint candidates. Then, it uses VGG for extracting part probability score maps and, finally, associates the joints with Non- Maximum Suppression.

2.2.2. OpenPose

OpenPose [12] is a real-time bottom-up approach for detecting 2D pose of multiple people on an image. First, it takes an RGB image (Figure 4: **OpenPose pipeline Reprinted from [4]a**) and apply a fine-tuned version of the convolutional neural network VGG-19 [13] to generate the input features of the algorithm. Second, these features enter a multi-stage CNN, created on the

basis of the convolutional pose machines developed in [14], to predict the set of confidence maps (Figure 4: **OpenPose pipeline Reprinted from [4]b**), where each map represents a joint, and the set of part affinities (Figure 4: **OpenPose pipeline Reprinted from [4]c**), which represent the degree of association between joints. Lastly, after a non-maximum suppression in order to discretize the data, bipartite matching is used to associate body part candidates (Figure 4: **OpenPose pipeline Reprinted from [4]d**) and obtain the full 2D skeletons (Figure 4: **OpenPose pipeline Reprinted from [4]e**), i.e. the joint that share the higher weight are selected between all the possible pairs of associations.

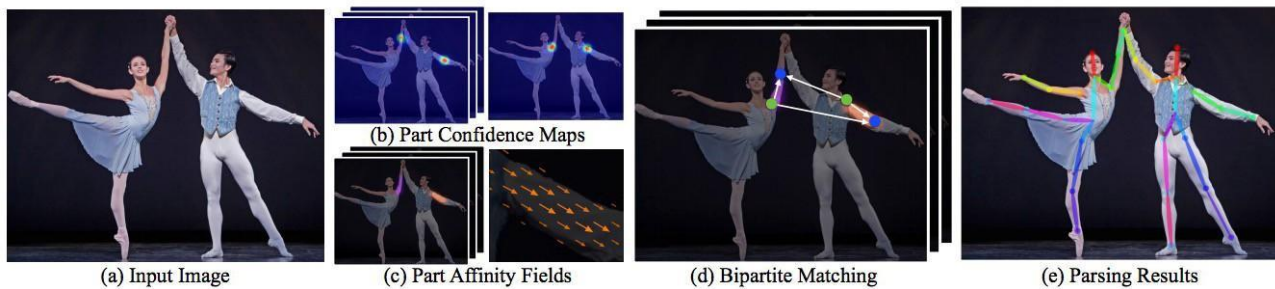


Figure 4: OpenPose pipeline Reprinted from [4] (source: IEEE MultiMedia, vol. 19, 2012)

2.2.2.1. Comparison between the frameworks

In our work, we have chosen to use OpenPose. The main reason is that OpenPose is a bottom-up approach, we avoid a possible early commitment, that includes a hand skeleton estimation, which has been considered an important perspective in our system. Table 1 shows a list of popular open source methods for 2D pose estimation, their Githubs, the machine learning framework used for the implementation and the classification obtained in their respective challenges.

Method	Git.	ML framework	Benchmark / Dataset	Rank	mAP (%)
AlphaPose	[5]	PyTorch	COCO Keypoint challenge 2018	11	70.2
DensePose	[7]	Caffe2	Posetrack multi-person pose estimation 2017	7	61.2
HMR	[9]	TensorFlow	Common objects in context (COCO)	–	–
DeepCut	[11]	Caffe	MPII Multi-Person dataset	–	51.4
OpenPose	[4]	Caffe	COCO Keypoint challenge 2016	1	60.5

Table 1: List of popular open-source frameworks (source: ICVS conference, 2019) [2]

2.3. Gesture recognition methods

The implementation of deep learning for gesture recognition has become the common practice and can lead to very good results. The ChaLearn LAP Large-scale Isolated Gesture Recognition Challenge from the ICCV 2017, crowned [15] [16] [17] as the best deep learning algorithms for gesture recognition. However, the need for large training databases is not compatible with the constraints professional gestures where datasets are quite small. Therefore, in this research we will study others gesture recognition methods.

2.3.1. Dynamic Time Warping (DTW)

Dynamic Time Warping (DTW) [18] as well as Hidden Markov Models [19] are two machine learning methods widely used in pattern recognition. DTW is a template- based approach that is based on a temporal re-scaling of a reference motion signal and its comparison with the input motion signal, i.e., it measures the similarity between two temporal sequences, such as in [20] where they use DTW for off-line recognition of a gestures. DTW is good for doing one-shot learning, because can detect similarity between two temporal sequences even that one is faster than the other, while HMMs is a robust duration-independent model-based approach.

2.3.2. Hidden Markov Models (HMMs)

Hidden Markov Models is a statistical model for modelling time series data based on the Markov chain or Markov property, i.e. each event depends only on the previous event ($P(X_t = j | X_1 = i_1, \dots, X_{t-1} = i_{t-1}) = P(X_t = j | X_{t-1} = i_{t-1})$). In HMM, we make two assumptions: first, the observation at time t comes from a hidden process state (S) and second, it satisfies the Markov property. Therefore, joining these two assumptions, with S_t the current state and Y_t the observed variable, we obtain:

$$P(S_{1:T}, Y_{1:T}) = P(S_1, Y_1) \prod_{t=2}^T P(S_t|S_{t-1})P(Y_t|S_t) \quad (2.2)$$

We chose to use the work described in [21], which makes use of K-means to model the time series of motion data and HMMs for classifying and recognizing the gesture classes by using the Gesture Recognition Toolkit (GRT)[22].

2.4. Evaluation framework and metrics

In this section, we will explain the metrics used for the evaluation of the gesture recognition results.

Confusion matrix: allows the visualization of the performance of a supervised learning algorithm relating the actual gesture to the predicted gesture.

		Actual Gesture	
		Gesture X	Gesture Y
Predicted Gesture (HMM)	HMM X	True Positive X	False Negatives Y
	HMM Y	False Negatives X	True Positive Y
		False Positive Y	

Table 2: Example of confusion matrix (source: Master Thesis, UAM Polytechnic School, 2019) [32]

Recall: in this research we define recall as the percentage of total gestures performed and correctly classified by the algorithm. Equation 2.3 shows the general and the applied equation of the recall measure.

$$\text{Recall}(Rc) = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negatives}} = \frac{\#(\text{gestures correctly recognized})}{\#(\text{gestures performed})} \quad (2.3)$$

Precision: in this research we define precision as the percentage of total gestures performed and correctly recognized by the algorithm. Equation 2.4 shows the general and the applied equation of the precision measure.

$$\text{Precision}(Pr) = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{\#(\text{gestures correctly recognized})}{\#(\text{gestures classified})} \quad (2.4)$$

Accuracy: we define accuracy as the percentage of total gestures performed, correctly recognized and correctly classified by the algorithm. Equation 2.5 shows the general and the applied equation of the accuracy measure.

$$\text{Accuracy}(Ac) = \frac{\text{True Positive} + \text{True Negative}}{\text{Total}} = \frac{\#(\text{gestures correctly recognized})}{\#(\text{gestures})} \quad (2.5)$$

F-score: defines the harmonic mean between the precision and the recall. Equation 2.6 shows the general equation of the f-score measure.

$$F \text{ score} = 2 \frac{\text{Pr} \times \text{Rc}}{\text{Pr} + \text{Rc}} \quad (2.6)$$

3. Data recording for Pose Estimation

3.1. Data recording using RGB-D cameras

The final goal of this task is to succeed real time body tracking by making use of cameras and pose estimation techniques. To record images, two different RGB-D cameras were used, each of them at one different recording session: the Intel RealSense Depth Camera D435 and the frontal camera (True Depth Camera) together with the rear camera (Dual Rear Camera) of the i-phone XS. A customized framework permitting to record these images from the i-phone has been developed and described in details in [2]. The capture setting presets have been defined to be similar for both devices in terms of image resolution, fps etc. so that the images recorded could be comparable and could be part of the same fused dataset despite the fact that they have been recorded with 2 different cameras. The principle reason for developing a particular framework permitting to record images with an i-phone and for using 2 different devices with similar settings is a gain in technological flexibility for museums and their users. The use of Smartphones is much broader and simpler than the specific RGB-D cameras and starts gaining place even in research areas.

Concerning the use of the RealSense camera, before doing the recordings it is necessary to have installed the software development kit (SDK) and the Intel RealSense viewer, a software that allows to select the camera presets and record the RGB-D frames.

Secondly, the RealSense camera saves the sequences in a unique ROS bag file extension, format created by the Robot Operating System (ROS). Therefore, it is necessary to perform a conversion to a format readable by the pose estimator (png or jpg). This conversion is done with the platform Intel rs-convert Tool.

Finally, a conversion must be performed for both images recorded with the RealSense and the i-phone, to get the RGB-D sequences with a resolution of 640x480 pixels and a frame rate of 30 fps; with which we can train and test the gestures recognition module.



Figure 5: Illustration of the 2 cameras used (source: www.intel.com, www.apple.com) [23], [24]

3.2. Pose estimation: 3d extraction from 2D provided by Open Pose

The goal of pose estimation is to obtain a series of keypoints that can be used by a gesture recognition engine as input, enabling to train a Machine Learning Model that adapts to different situations or environments. OpenPose, in our case, estimates 25 body keypoints, 2x21 hand keypoints and 70 face keypoints. However, some of the estimated keypoints are useless for the recognition, either they are occluded or they do not carry any information about the gesture. We select thus the joints of interest and a 3D model of the skeleton is created. In order to do this, a weighting is made between the values of the pixels obtained by OpenPose and the depth map, it means, if OpenPose returns the coordinate value x_{J_n} and y_{J_n} for the joint J_n with the depth map Z , we will take as depth value, z_{J_n} , the result of equation 3.5

$$z_{1J_n} = Z[\text{floor}(x_{J_n})][\text{floor}(y_{J_n})] \times (\text{ceil}(x_{J_n}) - x_{J_n}) \times (\text{ceil}(y_{J_n}) - y_{J_n}) \quad (3.1)$$

$$z_{2J_n} = Z[\text{floor}(x_{J_n})][\text{ceil}(y_{J_n})] \times (\text{ceil}(x_{J_n}) - x_{J_n}) \times (y_{J_n} - \text{floor}(y_{J_n})) \quad (3.2)$$

$$z_{3J_n} = Z[\text{floor}(x_{J_n})][\text{floor}(y_{J_n})] \times (x_{J_n} - \text{floor}(x_{J_n})) \times (\text{ceil}(y_{J_n}) - y_{J_n}) \quad (3.3)$$

$$z_{4J_n} = Z[\text{floor}(x_{J_n})][\text{floor}(y_{J_n})] \times (\text{ceil}(x_{J_n}) - x_{J_n}) \times (y_{J_n} - \text{floor}(y_{J_n})) \quad (3.4)$$

$$z_{J_n} = z_{1J_n} + z_{2J_n} + z_{3J_n} + z_{4J_n} \quad (3.5)$$

3.3. Gesture recognition

The joints obtained with the pose estimation and the data obtained from the depth camera are the input to the gesture classification algorithm. To make the recognition invariant to the position of each person in the frame, the neck joint has been taken as a reference point, and any frame without neck has been discarded.

The gesture recognition engine is based on supervised learning. Therefore, before making the gesture recognition, a labelled database has been manually created by manually selecting starting and ending time stamps of each gesture.

Once the database has been labelled, it is necessary to process it in order to organize it into logical groups. A .grt document has been created by concatenating in rows all the coordinates, 2D or 3D, of the different joints and the different frames of the each gesture. Moreover, we normalize the pixel values between 0 and 1.

Finally, the gesture recognition engine uses k-Means to obtain discrete-valued observations. Then, Hidden Markov Models is used to train the discrete data and to determine a gesture recognition accuracy. The platform GRT has been used for the entire process. Figure 6 shows the gesture recognition pipeline.

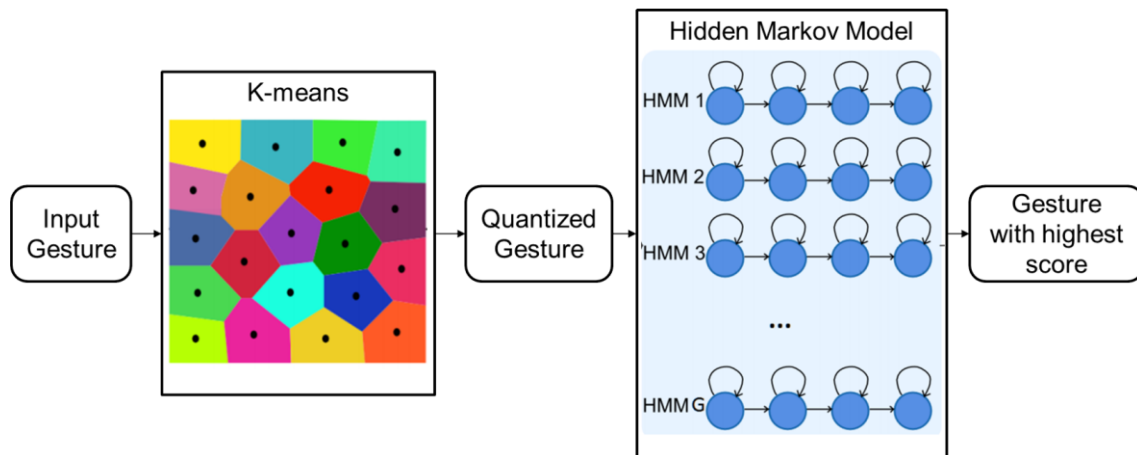


Figure 6: Gesture recognition pipeline followed by the application (source: Autonomous Robots) [21]

3.4. The datasets used for pose estimation and gesture recognition

3.4.1. The Glass Blowing dataset

The Glass Blowing dataset, has been recorded in a French Centre for research and training of glasswork (CERFAV). As described before 2 devices have been used to record RGB-D images of the glass master performing the necessary tasks and gestures for the creation of a jar. The procedure has been segmented into 6 gestures. Among these gestures the first 2 are considered to be preparatory since the glassblower prepares the material by using the furnace (G1) and then he moves the bar to soften the material (G2). Starting from the third gestures the glass master interacts directly with the material and starts the creation process. The craftsman executes the gestures in a very limited space since a specific metallic construction is used to support the pipe. This allows the artist to have a reference frame and to perform various manipulations of the glass by using also some professional tools (pliers, specific paper etc.). The majority of gestures are done while the craftsman is sitting. More precisely he starts by shaping the neck of the carafe with the use of pliers (G3), then he tightens the neck in order to define the transition between the neck and the curved vessel (G4), he holds in his right hand a specific paper and shapes the curves of the blown part (G5) and finalizes the object and fixes the details by using a metallic stick (G6). In general, the right hand is manipulating the tools while the left is holding and controlling the pipe. Table 3 shows the six gestures identified and labelled during the sequences.







Gesture 1 (G1)	Gesture 2 (G2)	Gesture 3 (G3)
		
Insert the glass into the furnace	Move the bar from one side to the other	Shaping glass with the hand
Gesture 4 (G4)	Gesture 5 (G5)	Gesture 6 (G6)
		
Blow through the stick	Tighten the base of the glass with the pliers	Burn the base of the glass with a torch

Table 3: Example of frames from Glass Blowing dataset (source: compiled by Mingei, 2019) [25]

The Glass Blowing dataset contains 13 samples from G1, 12 from G2, 23 from G3, 9 from G4, 16 from G5 and 5 from G6 performed by one glass master.

The final goal of recording these images is to estimate the body pose of the glass blower. As explained in the section 5.1.1 pose estimation is a challenging process since various technical difficulties may occur (listed in the 5.1.1). To overcome these difficulties, complementary recording have been performed for the gestures 3 to 6 from different point of views, by placing the camera at slightly different recording angles and distances from the subject, combining thus frontal and lateral point of views. This permits to face occlusion issues, to have a bigger variety in the data and to avoid overfitting phenomenon during the training of machine learning models that comes after

pose estimation. The table below presents some representative images recorded per gesture with 4 different camera's angles.

	Camera's angle 1	Camera's angle 2	Camera's angle 3	Camera's angle 4
Gesture 3				
Gesture 4				
Gesture 5				
Gesture 6				

Table 4: Example of frames recorded with different camera's angles in the Glass blowing dataset (source: compiled by Mingei, 2019) [25]

3.4.2. The Silk Weaving dataset

Finally, Silk Weaving dataset contains sequences recorded at the museum “Das Haus der Seiden kultur” in Krefeld, Germany. These images represent the weaving process by manipulating professional machines from XVIII century. Taking into consideration the scene configuration and after several tests, the best camera's set up permitting to have a clear access to weaver's body was defined to be the lateral view. Each sequence contains around 6.000 RGB frames with a resolution of 640x480 and a frame rate of 30fps.

In Silk Weaving dataset only one user has been recorded. Table 5 shows the three gestures identified and labelled during the sequences along with the skeleton of each of them. The dataset contains 88 repetitions of each gesture performed. For more information about the details of weaving process please refer to the deliverable 5.1.




Gesture 1	Gesture 2	Gesture 3
		
Press the treadle and push the batten	Move the shuttle sideways	Leave the treadle and pull the batten

Table 5: Example of frames from the Silk Weaving dataset (source: ICVS conference, 2019) [226]

3.4.3. The Mastic dataset

The process of mastic harvesting was the last one to be recorded among the 3 pilots of MINGEI. It has been recorded in September because of seasonal constraints of the cultivation calendar. The main goal of the video recordings done was mostly to assure a visual illustration of the harvesting process. The tracking of cultivators' body and the estimation of their pose can be done in offline mode but is not suitable in this project for a real time application since no interactive installation will be developed requiring real time body tracking. Real-time body tracking will be used in the Glass blowing pilot as explained in the deliverable 5.3.

More precisely in the Mastic Pilot video recordings were done under real conditions while the cultivators were performing their tasks in the field in the open-air museum area but also in a controlled environment where the cultivators simulated the gestures. The reason for doing these both recordings was linked to the fact that in open air recordings various factors may impact the quality of images such as the natural lightning conditions, the presence of other people, even museum's visitors in the frame etc. while in a controlled environment these risks are reduced: the number of people present in the room is limited etc.

From motion capturing point of view, mastic harvesting process is of a different nature from the 2 previous pilots. The gestures here are of bigger amplitude, while the entire body is actively involved. The cultivator is changing constantly his/her body postures from standing to kneeling or sitting and then standing up again. 8 gestural units that could be also considered as tasks have been identified here, as presented in the table 8. Similarly to the Silk Weaving recordings, the best position for camera recordings was found to be from the side of the expert (lateral view). This permitted to record cultivator's entire body and to have a clear representation of the task executed.









Gesture 1 Scrapping	Gesture 2 Sweeping	Gesture 3 Dusting	Gesture 4 Wounding
			
Gesture 5 Gathering	Gesture 6 Harvest from tree	Gesture 7 Wiping	Gesture 8 Shifting
			

Table 6: Example of frames from the Mastic cultivation dataset (source: compiled by Mingei, 2019) [27]

4. Evaluation

4.1. Comparative evaluation

In order to evaluate the performance of our pose estimation as well as the feature extraction and selection performed, different methods can be used such as the filter-based or wrapper based methods as the explained in [29]. In a filter-based approach a score is assigned to each feature based on how it allows to regroup observations corresponding to the same activity (low intraclass variance), and differentiate observations that correspond to different activities (high interclass variance [30]). From the other hand in the wrapper-based method the feature selection is done based on the recognition performance of a classifier [31]. The idea here is to identify the most suitable combination of features that provide the highest classification/ recognition score. In the experiment presented below the second method has been used.

More precisely in the gesture recognition pipeline the 80%-20% evaluation criteria has been employed. We randomly divide our dataset in 80% training set and 20% as testing set, repeating this procedure 10 times and computing the average values to generate the confusion matrix. We also use the Recall (Rc), Precision (Pc) and f-score metric.

Moreover, for the gesture recognition engine, 30 clusters for the K-Means algorithm and 15 states for the HMMs were selected, which follow an ergodic topology. Finally, different tests have been done in order to compare the gesture recognition accuracy using the following criteria: 2D (only X and Y) against 3D (X and Y + Z extracted from depth images), and 2 joints (left and right wrist) versus 7 joints (the neck, the right and left wrists, elbows and shoulders).

4.1.1. Pose estimation challenges

The gesture recognition engine receives as input the joints estimated by OpenPose, therefore, the recognition of gestures highly depends on the quality of the pose estimation. In this section, some of the most common problems encountered during the pose estimation are being analysed:

- Skeleton not estimated: the pose estimation algorithm does not detect any person in the frame and therefore does not return any joint. This situation usually occurs when the person continues performing their tasks outside the reach of the camera, or when the person is too blurred for being recognized.
- Skeleton incorrectly estimated: in this case, the algorithm detects and estimates the person in the image. However, the estimation is not correct. This is a high influential error since the generated input is erroneous and, therefore, the Hidden Markov Model will use erroneous estimation for training.
- Partial skeleton estimation: this is the case where, either because the image is too blurry, or because the person is not detected correctly, the estimated skeleton is incomplete, i.e. not all joints are estimated.
- Skeleton occluded: in line with the previous case, if part of the human body is occluded, either by an object, or because one part of the body occludes the other, the skeleton will not be completely estimated.

- Wrong person estimated: as Open Pose is a multiple people estimation framework, it may happen that the person interested in being estimated in the framework is not person with the highest score, being this, the discrete method to track the target person.

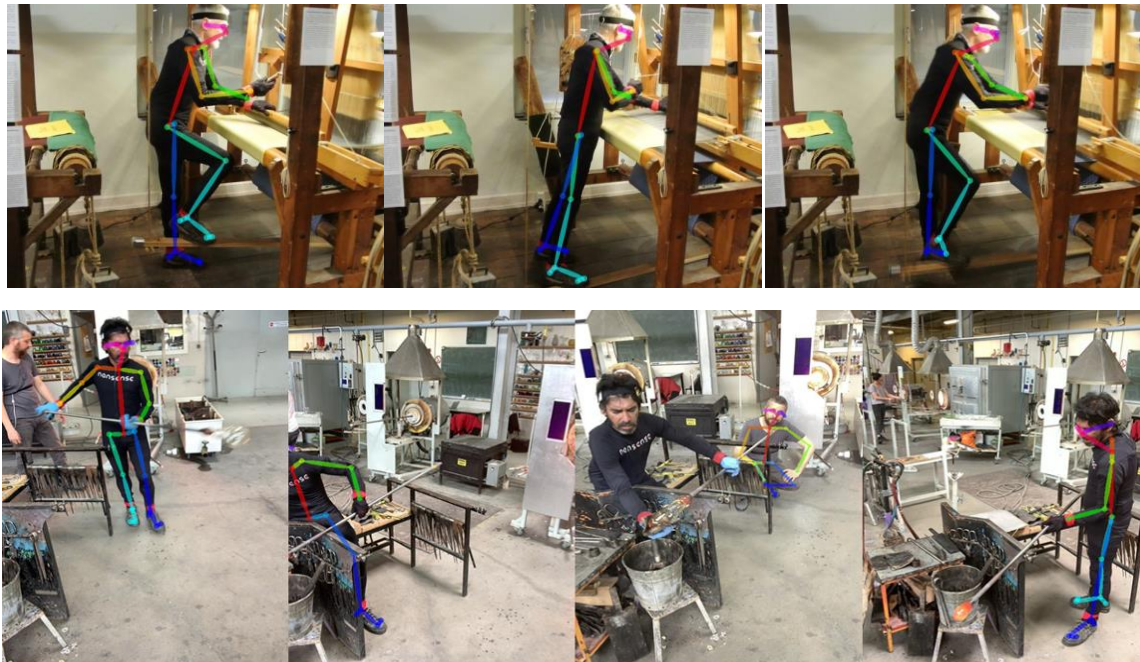


Figure 7: Example of skeleton estimation in Silk Weaving and Glass Blowing dataset (source: Mingei, 2019) [25], [26]

In Figure 7 we can see some examples of skeleton estimation in the two different datasets. From left to right and from top to bottom we are going to explain the type of estimation in that frame: (1) Frame perfectly estimated. (2) Frame perfectly estimated. (2) Frame perfectly estimated. (4) Frame perfectly estimated. (5) Frame with skeleton occluded. (6) Frame with the wrong person estimated. (7) Frame with skeleton occluded.

Gesture 1 Scrapping	Gesture 2 Sweeping	Gesture 3 Dusting	Gesture 4 Wounding
Gesture 5 Gathering	Gesture 6 Harvest from tree	Gesture 7 Wiping	Gesture 8 Shifting

Table 7: Examples of skeleton estimation in the mastic dataset (source: compiled by Mingei, 2019) [26]

In the table 7 we can observe examples of pose estimation of mastic cultivators. The general conclusion is that while the person is kneeling the algorithm doesn't provide an accurate estimation of leg's positions and more precisely calf's position (gathering, harvesting from tree). This might also be linked to the fact that the legs are positioned too close to each other because if we observe the sweeping gesture both legs are well detected. This issue is also due to occlusions, since in several cases one of the two legs remain hidden behind the other (scrapping). However as explained in the section 4.4 the priority is placed on the pose estimation in the glass blowing dataset. Pose estimation is less crucial in the Silk weaving and mastic pilots because they concern much less dexterous tasks and little manual skill. The results of body tracking tests have been presented in the 5.1.1 and 5.1.2 sections. For the three of them with a focus on a glass blowing and a small comparison between glass blowing and silk weaving that have such different configurations.

As explained in 4.4 to overcome some of the pose estimation difficulties especially in the Glass blowing dataset, complementary recordings from different point of views have been done. In the table below we can see visual examples of the body tracking results in these images. In most of these camera's angles the goal is to avoid occlusions. Attention has been paid to avoid situations where the subject moves out the camera's range, as it was the case in the initial recordings (figure 7).

Moreover to overcome the difficulty of multiple person detection the Z feature has been used by choosing the skeleton that is closest to the camera, thanks to the depth information. More specifically, during motion capturing with cameras, the X, Y, and Z coordinates of the people in front of the camera are being captured. Using the torso as the reference point, we get the Z coordinate of the torso of the person with the smallest value, as it is the one closest to the camera and this mechanism permits to avoid multiple person detection.













	Camera's angle 1	Camera's angle 2	Camera's angle 3	Camera's angle 4
Gesture 3				
Gesture 4				
Gesture 5				



Table 8: Example of pose estimation in images recorded with different camera's angles in the Glass blowing dataset (source: compiled by Mingei, 2019) [25]

4.1.2. Comparison of pose estimation results in 2 datasets

A comparison of the pose estimation using the two datasets has been made. We compared the number of frames per gestures with (1) the percentage of frames without any estimated skeleton, thus without estimation at all; (2) the percentage of frames without any reference point, thus without the neck and, (3) the percentage of frames having the minimum useful estimation, thus at least the neck.

The results are shown on Table 9 and we can affirm that the lateral views provided by the Silk Weaving dataset and all the views provides by Glass Blowing dataset have a good potential, since a full skeleton has been estimated for all the frames. We would also expect that the small duration of the G1 of Silk Weaving dataset might also affect its recognition accuracy. Moreover, from the duration of Glass Blowing we can notice how the duration of the frames varies depending on the gesture, G6 usually lasts 526 frames while G2 last 56 frames, we assume that the shortest gestures (G2 and G4) will tend to fail more than the long ones.

<i>Dataset</i>	<i>Silk Weaving</i>		
Gesture	G2	G3	G7
# Frames per sample	30.85	41.81	66.16
% Frames without any skeleton estimated	0	0	0
% Frames without reference point (without neck)	0	0	0
% Frames minimum useful estimation (at least the neck)	100	100	100

<i>Dataset</i>	<i>Glass Blowing</i>					
Gesture	G1	G2	G3	G4	G5	G6
# Frames per sample	448.92	56.00	466.69	92.44	235.25	526.60
% Frames minimum useful estimation (at least the neck)	100	100	100	100	100	100

Table 9: OpenPose results on the Silk Weaving and Glass Blowing datasets (source: Master Thesis, UAM Polytechnic School, 2019) [32]

As explained in the section 4.4 in this task the priority is placed on the pose estimation in the glass blowing dataset. Pose estimation is less crucial in the Silk weaving and mastic pilots because they

concern much less dexterous tasks and little manual skill. However the results of body tracking tests have been presented in the 5.1.1 and 5.1.2 sections. for the three of them with a focus on a glass blowing and a small comparison between glass blowing and silk weaving that have such different configurations and contexts.

4.1.3. Gesture recognition with data from Silk Weaving dataset

To select features for the Silk Weaving dataset, given the successful pose estimation (discussed in 5.1.2) we formulate the hypothesis that the simplest combination of features might permit to achieve a successful gesture recognition. Thus gesture recognition of silk weavers is done by using the algorithm presented in the section 4.3. The accuracy in the Table 10 is 100%, meaning that the recognizer works perfectly for the gestures of the Silk Weaving dataset when using as input positions of only 2 joints (wrists) in only 2 dimensions (X and Y). The three gestures performed in the Silk Weaving are different between them facilitating thus the estimation of recognition probabilities. We can also observe that these gestures have an important amplitude and variance in space. They are mostly tasks/actions executed by the whole weaver's body and not fine hands' or fingers' movements.

2J-2D	G1	G2	G3	Pr(%)
<i>HMM1</i>	183	0	0	100
<i>HMM2</i>	0	166	0	100
<i>HMM3</i>	0	0	181	100
Rc(%)	100	100	100	100 ± 0

Table 10: Gesture recognition results using 2 joints and 2 dimensions on Silk Weaving dataset (source: ICVS conference, 2019) [2]

4.1.4. Gesture recognition comparisons using the Glass blowing dataset: 2d vs 3d, 2 joints vs 7 joints

As far as the Glass Blowing dataset is concerned another hypothesis was expressed, that the Z feature corresponding to the 3rd dimension, could permit to achieve better recognition results since it would complete the other 2 (X and Y) features. According to the results presented in table 8 the hypothesis is confirmed. The final accuracy obtained in the 2D case is 58.7% far surpassing a random recognizer (16.66%), while if we use the 3D data we improve the recognizer up to 62.5% accuracy rate. This last statement shows that the experiment carried out to add the depth maps is beneficial in the gestures recognition. This experiment has been done on the dataset containing 6 gestures, including the preparatory ones, because especially the first gestures present a bigger spatial variability and it is interesting to observe how does the 3rd dimension impact the recognition results. Even if globally the 3rd dimension permits to improve the total accuracy by 4% , the difference in the accuracy when using 2d or 3d is still small.

Analysing in more detail the results of the Table 11, we can notice that G6 is always the worst-recognized gesture, we assume this is because there are only 5 samples of this gesture, on several occasions G6 is not even selected for testing, in contrast, the gestures with more samples, G1, G3 and G5, are the ones with the highest precision.

2D	<i>G1</i>	<i>G2</i>	<i>G3</i>	<i>G4</i>	<i>G5</i>	<i>G6</i>	Pr(%)
<i>HMM1</i>	14	11	0	0	0	0	56
<i>HMM2</i>	0	25	0	0	0	0	100
<i>HMM3</i>	0	11	25	4	1	2	58
<i>HMM4</i>	0	0	1	17	0	0	94
<i>HMM5</i>	2	8	2	9	1 1	2	32
<i>HMM6</i>	2	2	4	3	2	2	13
Rc(%)	77	43	78	51	78	33	58.7 ± 8.9
3D	<i>G1</i>	<i>G2</i>	<i>G3</i>	<i>G4</i>	<i>G5</i>	<i>G6</i>	Pr(%)
<i>HMM1</i>	9	19	0	0	0	0	32.1
<i>HMM2</i>	1	26	0	0	0	0	96.3
<i>HMM3</i>	0	11	34	0	0	0	75.6
<i>HMM4</i>	0	0	0	22	0	0	100.0
<i>HMM5</i>	2	7	1	4	1 3	0	48.1
<i>HMM6</i>	0	3	5	3	0	0	0.0
Rc(%)	75	39	85	75	100	0	62.5 ± 6.3

Table 11: Gesture recognition results comparison between 2D versus 3D for 2 joints in the glass blowing dataset (source: Master Thesis, UAM Polytechnic School, 2019) [32]

Another experiment was conducted to see how does the number of joints used to train the gesture recognition engine impact the recognition accuracy. More precisely a comparison has been done between two joints (right and left wrists) and seven joints (neck, right and left wrists, right and left shoulders, and right and left elbows). The glassblower seems to move his whole upper body and mostly his hands. A comparison of the recognition results can be found in Table 12,13.

2joints	<i>G1</i>	<i>G2</i>	<i>G3</i>	<i>G4</i>	Pr(%)
<i>HMM1</i>	26	2	0	1	89
<i>HMM2</i>	0	19	1	1	95
<i>HMM3</i>	7	9	16	4	44
<i>HMM4</i>	2	4	4	22	68
recall (%)	74	56	76	81	

Table 12: Gesture recognition results when using 2D for 2 joints and 4 classes instead of 6 (source: Diploma thesis, AUTH University, 2020) [31]

7joints	<i>G1</i>	<i>G2</i>	<i>G3</i>	<i>G4</i>	Pr(%)
<i>HMM1</i>	31	0	2	0	93
<i>HMM2</i>	2	33	2	0	89
<i>HMM3</i>	1	1	16	0	88
<i>HMM4</i>	1	0	1	27	93
recall (%)	88	97	80	100	

Table 13: Gesture recognition results when using 2D for 7 joints (source: Diploma thesis, AUTH University, 2020) [31]

According to the tables 12 and 13 the precision was improved for all gestures apart from G3 by 2% and up to 23%, which is also the case for the recall, which is improved by up to 25% for G4, proving that the elbow and shoulders were important for the HMMs to achieve a good recognition performance. Table 14 shows the total accuracy and mean f-score for the experiments with two and seven joints below and is also very indicative of the result.

Metric	<i>2 joints</i>	<i>7 joints</i>
<i>f-score</i>	70%	90%
<i>total accuracy</i>	71%	91,5%

Table 14: Mean f-score and total accuracy for the glassblowing dataset with 2 joints and 7 joints (source: Diploma thesis, AUTH University, 2020) [31]

When extracting 7 joints from the videos, the mean f-score together with the total accuracy are improved by 20%, also confirming the importance of different limbs of the upper body.

4.2. Conclusion

In this section we have experimented with 2 different datasets, Silk Weaving and Glass Blowing dataset, in 2D or 3D, when using 2 joints and 7 joints. The best results in gesture recognition have been obtained with the Silk Weaving dataset showing that in terms of feature selection using information only from 2 joints in 2 dimensions is sufficient. In the glass blowing dataset the 3rd dimension seems to improve slightly the results while an important difference is detected when using 7 joints instead of only 2.

More precisely these observations permit to conclude that:

1. The use of the 3rd dimension brings an added value however the difference of results between 2D and 3D doesn't seem to be that big. In practice it means that working only with 2 dimensions could be sufficient. However if the cost (in terms of computation power and time, of technical issues etc.) of extracting the 3rd dimension is not high, then it could still be interesting to continue measuring it in future experiments and having it as a complementary feature. Another parameter that impacts the importance of the depth information is the nature of the gesture itself. In extremely fine movements where the articulations do not move a lot on the Z axis, the depth information might constitute also a source of noise instead of being a valuable feature. From the other hand, in bigger, more ample gestures, where human body or a precise joint moves and the depth distance to the camera varies, it can represent an interesting complementary feature.
2. The use of features extracted from 7 joints, the whole upper body, instead of 2, the wrists seems to improve considerable the recognition results meaning that the articulations of the upper body play an important role during the execution of the gestures. When using only 2 joints, the incoming information could be considered as incomplete.

5. Conclusions and Future Work

This deliverable presents a) the motion visualization module that will be used for the representation of the craft as it was performed by the expert with the use of 3D avatars and b) the preliminary work done for the pose estimation that will be used as an input for the visitor's gesture recognition.

Several steps have been implemented so far to test and achieve a real time body tracking such as:

- Videos/images recordings of the 3 pilots and creation of the datasets
- Post processing of the videos/images by extracting the information about the 3rd dimension
- Comparative evaluation of different features in 2 datasets, by using gesture recognition accuracy results that permits to reveal the difference between the gestures performed in the pilots.

First experimental results of offline gesture recognition in 2 different datasets, Silk Weaving and Glass Blowing dataset, in 2D or 3D are presented. The best results in gesture recognition have been obtained with the Silk Weaving dataset, while Glass Blowing dataset is more complex and requires a more complete feature extraction.

The future work will be focused on performing the real-time pose estimation and facing the various technical and scientific difficulties that may occur (computational power, latency, occlusions etc.) However, in parallel an alternative method based on a weakly supervised End-to-End deep learning approach will be used to explore gesture recognition without pose estimation, by using directly the optical flow of incoming images. The effort will be focused on achieving satisfying results in real-time since the real-time processing of images is necessary for the interaction of the user with the final installation. The work done in the T5.2 and described in the D5.2 will be used as input to the work of the T5.3 (D5.3) focusing on the gesture modelling, recognition and comparison.

PART B - References

1. “Capturing depth in iphone photography, wwdc 2017.” <https://developer.apple.com/videos/play/wwdc2017/507/>.
2. Moñivar, P. V., Manitsaris, S., & Glushkova, A. (2019, September). Towards a Professional Gesture Recognition with RGB-D from Smartphone. In *International Conference on Computer Vision Systems* (pp. 234-244). Springer, Cham.
3. Intel, “Intel realsense depth camera d400-series,” 2017.
4. Z. Zhang, “Microsoft kinect sensor and its effect,” *IEEE MultiMedia*, vol. 19, pp. 4–12, April 2012.
5. “Alphapose.” <https://github.com/MVIG-SJTU/AlphaPose>.
6. S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” *CoRR*, vol. abs/1506.01497, 2015.
7. “Densepose.” <https://github.com/facebookresearch/DensePose>.
8. T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” *CoRR*, vol. abs/1612.03144, 2016.
9. A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, “End-to-end recovery of human shape and pose,” *CoRR*, vol. abs/1712.06584, 2017.
10. M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, “SMPL: A skinned multi-person linear model,” *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, vol. 34, pp. 248:1–248:16, Oct. 2015.
11. L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, “Deepcut: Joint subset partition and labeling for multi person pose estimation,” 2015.
12. Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. Sheikh, “Openpose: Realtime multi-person 2d pose estimation using part affinity fields,” 2018.
13. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014.
14. S. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” *CoRR*, vol. abs/1602.00134, 2016.
15. L. Zhang, G. Zhu, P. Shen, and J. Song, “Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition,” *ICCV workshop*, 2017.
16. H. Wang, P. Wang, Z. Song, and W. Li, “Large-scale multimodal gesture recognition using heterogeneous networks,” *ICCV 2017 Workshop*, pp. 3129–3137, 2017.
17. P. Wang, W. Li, S. Liu, Z. Gao, C. Tang, and P. Ogunbona, “Large-scale isolated gesture recognition using convolutional neural networks,” 2017.
18. *Dynamic Time Warping*, pp. 69–84. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.
19. L. Rabiner and B. Juang, “An introduction to hidden markov models,” *ASSP Magazine, IEEE*, vol. 3, pp. 4–16, Jan. 1986.
20. A. Corradini, “Dynamic time warping for off-line recognition of a small gesture vocabulary,” pp. 82–, 2001.
21. E. Coupeté, F. Moutarde, and S. Manitsaris, “Multi-users online recognition of technical gestures for natural human–robot collaboration in manufacturing,” *Autonomous Robots*, Feb 2018.
22. N. Gillian and J. A. Paradiso, “The gesture recognition toolkit,” *Journal of Machine Learning Research* 15, 2014.
23. <https://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html>

24. <https://www.apple.com/>
25. D. Menychtas, B.E. Olivas, G. Senter, (2019) Frames from Glass Blowing dataset, Bibliographical reference: Unpublished research work.
26. D. Menychtas, B.E. Olivas, G. Senter, (2019) Frames from Silk Weaving dataset, Bibliographical reference: Unpublished research work.
27. D. Menychtas, B.E. Olivas, G. Senter, (2019) Frames from Mastic dataset, Bibliographical reference: Unpublished research work.
28. I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," Journal of machine learning research, vol. 3, no. Mar, pp. 1157–1182, 2003.
29. R. O. Duda, P. E. Hart, and D. G. Stork, Pattern classification. John Wiley & Sons, 2012.
30. R. Kohavi and G. H. John, "Wrappers for feature subset selection," Artificial intelligence, vol. 97, no. 1-2, pp. 273–324, 1997.
31. G. Senter, C. Kotropoulos, "Hybrid machine learning models for action recognition and trajectory forecasting", Diploma thesis, MSc in Digital Media & Computational Intelligence, Aristotle University of Thessaloniki Faculty of Sciences, Department of Informatics, Thessaloniki 2020.
32. Moñivar, P. V., Manitsaris, S., Cano, J., B., "Computer Vision for Body Tracking in Professional Environments", Master Thesis, UAM Polytechnic School (Escuela Politécnica Superior UAM), Madrid 2019.