Artificial Intelligence Research and Development
A. Cortés et al. (Eds.)

doi:10.3233/FAIA220338

Garment Manipulation Dataset for Robot Learning by Demonstration Through a Virtual Reality Framework

Arnau BOIX-GRANELL [a,1], Sergi FOIX [a] and Carme TORRAS [a]
[a] *Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, Spain*

Abstract. Being able to teach complex capabilities, such as folding garments, to a bi-manual robot is a very challenging task, which is often tackled using learning from demonstration datasets. The few garment folding datasets available nowadays to the robotics research community are either gathered from human demonstrations or generated through simulation. The former have the huge problem of perceiving human action and transferring it to the dynamic control of the robot, while the latter requires coding human motion into the simulator in open loop, resulting in far-from-realistic movements. In this article, we present a reduced but very accurate dataset of human cloth folding demonstrations. The dataset is collected through a novel virtual reality (VR) framework we propose, based on Unity's 3D platform and the use of a HTC Vive Pro system. The framework is capable of simulating very realistic garments while allowing users to interact with them, in real time, through handheld controllers. By doing so, and thanks to the immersive experience, our framework gets rid of the gap between the human and robot perception-action loop, while simplifying data capture and resulting in more realistic samples.

Keywords. Garment manipulation, learning by demonstration, virtual reality framework, cloth folding dataset

1. Introduction

Non-rigid object manipulation has gained a lot of attention during the last decade since it has proven to be one of the big milestones to reach in the field of robotics in order to come closer to achieving full human-like capabilities. But the robotic manipulation of deformable objects is certainly not an easy task. There are two main difficulties that robots must face when manipulating a deformable object. On the one hand, there is the problem of fully estimating its state. Due to their ability to deform, non-rigid objects can take an infinite amount of configurations in space. Since fully observability is impossible to have in a real scenario, estimations must be made. Whereas rigid objects' pose can be easily estimated once a portion of its body is identified and located in 3D space, the correct deformable objects' state is nearly impossible to detect with just partial observability. On the other hand, there is the problem of gracefully manipulating a deformable object for fulfilling a task. Among others, factors such as the friction, elasticity and thickness of the

[1]Corresponding Author: Arnau Boix-Granell, Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain; E-mail: arnau.boix@estudiantat.upc.edu

(a) (b)

Figure 1. The manipulation is done on top of a real table that has its virtual version inside of the framework. To make sure that both of the objects are located in the same space (virtual and real worlds) we used *HTC* trackers for extrinsic calibration. (a) Real setup showing HTC's tracker (top left), headset (middle) and controllers (bottom). (b) Virtual setup showing HTC's tracker (top left), controllers (bottom) and simulated garment (center).

fabrics, the weight, size and shape of the garment, determine, not only the possible type of grasping, but also which actions can be taken and which ones not. Probably, due to these difficulties, there are not as many good datasets of deformable objects as there are of their rigid counterparts. This fact slows down the development of new artificial intelligence algorithms capable of understanding this type of objects, and therefore, creates a knowledge gap that this work pursues to fill.

One of the main challenges when trying to develop a garment-based dataset is whether to use real pieces of fabric or to use simulated ones. Currently, most of the available datasets are based on RGB-D images coming from real clothing data [1–8]. Despite the convenience of having real data, it is very hard to extract the ground truth information from garments and humans during a manipulation sequence. Moreover, data tend to have noise and multiple occlusions, and post-processing is always needed in order to have good estimated labeling. On the other hand, other approaches exploit the use of simulation environments to easily obtain fully observable ground truth data, although they must program the cloth manipulation behaviours with scripts. Therefore, this type of data lacks human-like demonstrations, losing the crucial manipulation dexterity contributions that would be provided by having the human perception into the loop. Imagine, for instance, the movement followed by a human hand previous to the prehension of a deformable object. That trajectory will, first, determine whether the grasping point will be successful or not and, second, which are going to be the next possible actions over that object in order to fulfill the assigned task. Recall that deformable objects may change their state after a manipulation and that, depending on that action, that change may be irreversible without adding extra manipulations.

In order to overcome those challenges, we propose a new approach that combines the use of simulated garments with human-based manipulation trajectories. Thanks to a virtual reality (VR) framework, humans can interact in real time with simulated pieces of

(a) 1 PP^{c} (b) 1 LL^{m} (c) 2 PP^{c+c} (d) 2 PP^{m+m} (e) 2 PP^{c+m}

Figure 2. Classification of the different types of garment manipulations studied in this work: (a) One corner double point grasp (PP^{c}) with extrinsic planar contact (Π_e), (b) one middle edge double line grasp (LL^{m}) with (Π_e), (c) two corner double point grasp (PP^{c+c}) with (Π_e), (d) two middle edge double point grasp (PP^{m+m}), and (e) one corner, one middle edge double point grasp (PP^{c+m}) with (Π_e).

cloth (see Fig. 1a). In this work, we present a reduced but very accurate dataset of human cloth folding demonstrations.[2]

This article is structured as follows. Section 2 analyzes the related work in the literature. In Section 3 we present the different parts of our virtual reality set-up. Section 4 defines and explains our dataset storage format as an XML file. Section 5 introduces the different garments used in our experiments and the way we have classified them. Finally, Section 6 concludes this article.

2. Related Work

During the last decade, thanks to the creation of cloth manipulation datasets, a lot of progress in garment state detection and classification has been made. These datasets use either real or simulated fabrics in order to provide rich, and as accurate as possible, data reservoirs of cloth types, manipulation actions and garment states distribution.

2.1. Garment Datasets

In the context of garments, several attempts have been made to create various datasets. Some of those classify the garments by type [9–13], studying only static properties. Therefore, not useful when trying to understand manipulation processes. Others focus on the actions performed by a human when manipulating garments [14, 15]. Those works are mainly centred in studying the actions rather than the states of the piece of fabric, and for that reason, may not be as useful when trying to understand the evolution of garments between folding sates. Others use RGB-D (or RGB) images to perceive the distribution of the garment [1–8]. These approaches have to estimate the occluded parts of the piece of fabric and, for that reason, might not be as helpful when high precision methods are required.

At the time of the writing, and despite the broad variety of approaches, the authors have no knowledge of any other studies that provide both the actions developed by a human while manipulating garments and, at the same time, the tracking the full evolution of the piece of fabric from an original state (before manipulation) to an ending state (after manipulation). As previously stated, our approach aims to fill this void.

[2]The dataset can be found in: http://www.iri.upc.edu/groups/perception/clothingDataset/Data.rar

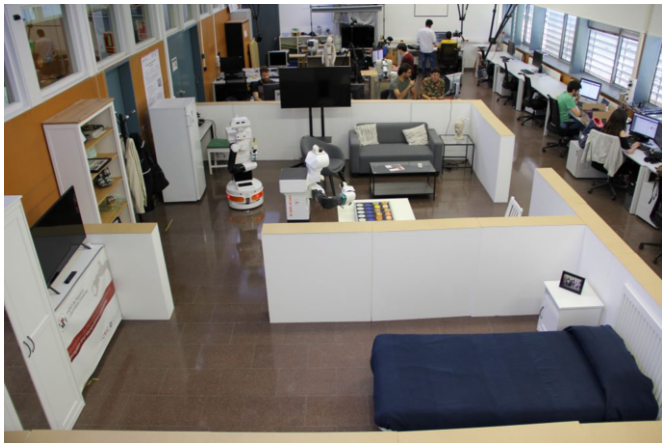

Figure 3. Perception and Manipulation Lab's apartment mock-up.

2.2. *Cloth State Manipulation*

A problem encountered when starting to develop the dataset was to define a proper way to classify the different cloth states during a manipulation. As we already know, garments can have an infinite number of configurations, and, consequently, an infinite number of possible manipulations can be applied to them. In order to be able to plan a sequence of actions to take a deformable object from one state to another one, we must simplify the state-action representation. For that reason, some researchers have classified the types of manipulation based on both cloth and grasp type attributes, such as type of contact (single point, linear or planar), number of grippers used (single handed or bi-manual), or its final manipulation state. Some examples of these classifications can be found in [16].

For this work, we have classified the manipulations depending on the number of grippers used (one or two), the type of contact (single point P, linear L, or planar Π) and the part of the garment where the contact is made. We have used a classification method similar to the one showed in [17] (See Fig. 2).

Despite having chosen this method, due to the full observability properties and the recorded ground truth information of the data, any other type of garment manipulation classification could be applied. This has been one of the reasons for developing this framework, providing the community with a tool to test and compare different classification methods, given that we believe that the value for each classification method depends on the manipulation task performed.

3. Set-up

This work has been developed in the Perception and Manipulation Laboratory at the Institut de Robòtica i Informàtica Industrial (CSIC-UPC), using an *HTC Vive Pro* headset for creating an immersive VR experience thanks to the scenes created under the *Unity* framework (see Fig. 1). The Perception and Manipulation Lab hosts a life-scale mock-up of a fully-equipped apartment (see Fig. 3). Inside the apartment, researchers can study the interaction between robots and users in close-to-reality domestic environments.

3.1. HTC Vive Pro

As previously stated, we wanted to develop a framework that not only allows the visualization of garments, but also allows to create realistic garment manipulations. In order to do so, we believed that virtual reality could create the desired interactive experience. With that objective, we used an *HTC Vive Pro*, a virtual reality headset developed by *HTC Corporation* in collaboration with *Valve Corporation*. This type of devices are famous for offering the possibility of entering into an immersive experience in which the user is able to interact into Virtual Reality (VR), Mixed Reality (MR) or Augmented Reality (AR) worlds.

Despite of its main use, *HTC* headsets are also used by developers in other fields, for example, several research teams are developing AR applications to enhance the learning of manual assemblies [18]. Besides the headset, the *HTC Corporation* also offers some accessories that help making a more immersive experience. The two devices used in this project are the *HTC Tracker* and the *HTC Controller* (see Fig. 1). The tracker eases the connection between the real and the virtual world, making it possible to connect virtual objects with its real counterpart (as long as it has the tracker attached). The controller not only sends its real position to the virtual world but it can also send some basic information using its integrated buttons. More precisely, the *HTC Vive's* controller offers a total of three different buttons, one pressure-sensitive trigger and a trackpad.

The *HTC* hardware can easily be connected to *Unity* downloading *SteamVR*'s application and its asset (downloadable from *Steam*'s store and *Unity*'s asset store, respectively). The asset implements basic prefabs that allow the creation of VR experiences where all of *HTC* components can be used.

3.2. Unity

For the development of the framework, we decided to use the *Unity* engine. *Unity* is a cross-platform game engine developed by *Unity Technologies* [19]. *Unity*'s engine is mainly used for game developing due to its versatile and easygoing interface. Despite of that, it is also used for several engineering and AI applications. For example, the implementation of intelligent agents capable of overcoming obstacles or solving basic games such as mazes or arcade-like games [20–23].

In this work, *Unity* is used to build a framework where the information coming from the HTC Vive Pro system is displayed in a 3D environment. Moreover, the game engine will also work as a data reading and processing tool. Out of the possible simulators that could have been used to develop this work *Unity* was chosen for having fast simulation and providing a flexible control [24]. On top of that, the game engine was also used because of its user-friendliness, allowing future research groups interested in this framework to reproduce it or to apply their own changes to the simulations, if desired.

3.2.1. Obi Cloth Unity asset

Once a simulation engine was chosen, the next step was to study how to simulate the garments within the 3D environment. After some research into the different asset extensions for cloth simulation within *Unity*, we discovered a dedicated collection of particle-based physics plugins for deformable objects, such as cloth, fluids, ropes and soft-bodies, called Obi [25]. Every Obi object is made by a set of particles that can interact with each

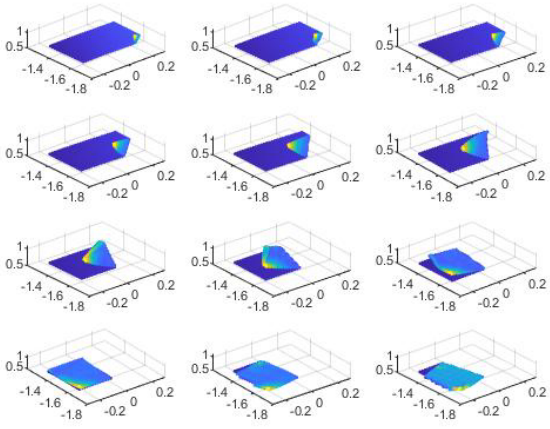

Figure 4. Sequence of point clouds obtained during a $PP^c + \Pi_e$ manipulation, with a z-axis heat map.

other, and affect or be affected by other objects within the scene. Moreover, particles can be constrained to have a customized behaviour. Compared to the other physics systems available in *Unity*'s assets store, Obi Cloth asset goes one step further by allowing much more constraints per cloth and by setting each particle's restriction separately.

4. Methodology

This section provides a brief explanation on how the dataset has been collected. Each manipulation is stored in a XML document with three main fields: Name, Mesh and Frames. The Name field corresponds to a string representing the name's experiment that has been performed. The Mesh field indicates the index of all the vertices that create a mesh element. Finally, the Frames field stores the evolution of the data at each timestamp.

For this dataset, we wanted to keep track of all the elements involved in a cloth manipulation task. In our current experiments, four elements were completely tracked. The first one, the garment per se. The dataset collects the coordinates of each particle of the fabrics, saving it under the tag name of *vertices*, inside of the geometry field within each frame (see Fig. 4). In order to easily export each mesh frame, data has been recorded maintaining *Ogre*'s mesh XML data-structure [26]. Secondly, we wanted to keep track of each of the *HTC* controllers used for manipulating the garments. In the case of a bi-manual operation, the tag names for each controller are *ControllerRight* and *ControllerLeft*. For each controller we store its pose components (position and rotation), a variable telling whether a grasping point is being held and a variable tracking the state of the trigger. This value was added thinking about future upgrades where changing the pressure over the surface of the grasped objects could be necessary for carrying out tasks such as edge tracing. Thirdly, it is also important to keep track of the position and rotation of the grasping points. Besides from that, and similarly to the controllers, we added a variable that indicates whether the right or left controller is holding the object or not. That third object type can be found under the tag of *GripPoint[i]*, where *i* is an integer value

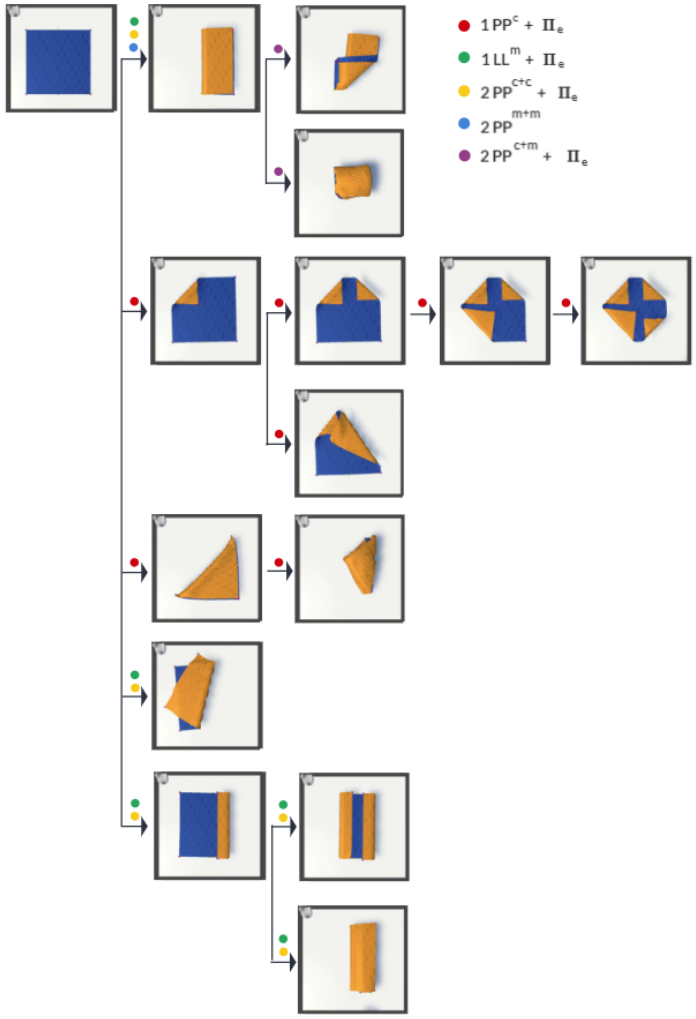

Figure 5. Graph of possible states for the napkin garment classified by manipulation using [17] method. The coloured dots indicate the different types of manipulation that can be performed in order pass from the previous state to the next one.

between one and the number of total simulated grasping points. Finally, the last object added to the dataset is the simulated table. From this object, the position and rotation are recorded under the name of Table.

All the experiments within the dataset have been conducted by a human using the HTC controllers as grippers. All the stated variables are recorded in XML files at 10 Hz. Finally, the dataset comes with two *.txt* files explaining both the format template of the XML documents and the data subsets distribution format.

5. Experiments

For a better versatility of the collected data, the conducted experiments have been divided into states. Each experiment starts in one described state and ends into another. With that methodology, if data of new experiments were required, only the new states would have to be recorded, given that the processes that follow the same sequence of actions can be reused.

A total of three different garments were used in these experiments, the properties of which can be seen in Table 1. These garments have been extracted from the household cloth object set studied in [27].

Table 1. Types of garment used in the experiments.

Name	Size [m]	Weight [kg]
Small Towel	0.3 x 0.5	0.08
Napkin	0.5 x 0.5	0.05
Tablecloth	0.90 x 1.30	0.188

In order to keep the dataset as brief and as rich as possible, we tried to just perform the most representative garment manipulations that are equivalent to all the studied garments. We use both single handed and bi-manual interactions, and we used them over different combinations of point, line and plane contact types. Due to the data format, it is easy to filter the manipulations by contact or interaction types with the objective of applying learning algorithms. Fig. 5 shows the complete sequence of states that have been studied for the *Napkin* case. As shown in the graph, some states can be achieved by performing different types of manipulation. For this garment, a total of nineteen manipulation sequences have been performed, with three repetitions each. These sequences correspond to all of the possible combinations of manipulations that start with the top-left state of Fig. 5 and end with one of the states on the right of the image. Whereas the *Small Towel* garment shares nearly the same state transition diagram, the *Tablecloth* garment is far too big for carrying on the examples within the state transition diagram. Despite that, we have included into our dataset a special case where the *Tablecloth* garment is hanging from a bar and has to set on the table thanks to a bi-manual manipulation and by taking advantage of the dynamics of the fabrics (see Fig. 6).

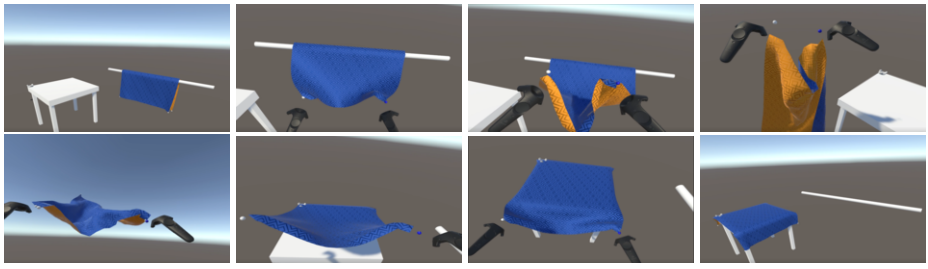

Figure 6. Manipulation of Tablecloth: From initial hanging position (top-left) to set on the table (bottom-right). The images in-between show frames from the two corner double point grasp manipulation performed on the Tablecloth garment.

6. Conclusions

In this work, we presented a *Unity* virtual reality framework to perform garment manipulation experiments. The approach used on the development differs form others in that we not only perform a full-mesh tracking but we also keep track of the position, rotation and interactions of other key features of the manipulation (like grippers or grasping points). Moreover, the implementation of the virtual reality allows the creation of an immersive experience that gets rid of the gap between the human and robot perception-action loop.

Later, we use the developed framework to create a rectangular garment manipulation dataset which is divided in states to allow a more versatile study. This new dataset aims to help the garment manipulation AI community by providing more realistic human-like garment manipulation data, which can be used in learning-from-demonstration approaches.

As a future work, we are planning on implementing a way to keep track of the states of the garments, by providing data such as how many corners are folded or if part of the garment is on top of another.

Acknowledgments

This work was developed in the context of the project CLOTHILDE ("CLOTH manIpulation Learning from DEmonstrations") which has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 741930) and is also supported by the BURG project PCI2019-103447 funded by MCIN/ AEI /10.13039/501100011033 and by the "European Union".

References

[1] Kimitoshi Yamazaki and Masayuki Inaba. Clothing classification using image features derived from clothing fabrics, wrinkles and cloth overlaps. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2710–2717. IEEE, 2013.

[2] Ioannis Mariolis and Sotiris Malassiotis. Matching folded garments to unfolded templates using robust shape analysis techniques. In *International Conference on Computer Analysis of Images and Patterns*, pages 193–200. Springer, 2013.

[3] Andreas Doumanoglou, Andreas Kargakos, Tae-Kyun Kim, and Sotiris Malassiotis. Autonomous active recognition and unfolding of clothes using random decision forests and probabilistic planning. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 987–993. IEEE, 2014.

[4] Gerardo Aragon-Camarasa, Susanne B Oehler, Yuan Liu, Sun Li, Paul Cockshott, and J Paul Siebert. Glasgow's stereo image database of garments. *arXiv preprint arXiv:1311.7295*, 2013.

[5] Bryan Willimon, Ian Walker, and Stan Birchfield. A new approach to clothing classification using mid-level layers. In *2013 IEEE International Conference on Robotics and Automation*, pages 4271–4278. IEEE, 2013.

[6] Georgies Tzelepis, Eren Erdal Aksoy, Júlia Borràs, and Guillem Alenyà. Semantic state estimation in cloth manipulation tasks. *arXiv preprint arXiv:2203.11647*, 2022.

[7] Arnau Ramisa, Guillem Alenya, Francesc Moreno-Noguer, and Carme Torras. Learning rgb-d descriptors of garment parts for informed robot grasping. *Engineering Applications of Artificial Intelligence*, 35:246–258, 2014.

[8] Enric Corona, Guillem Alenya, Antonio Gabas, and Carme Torras. Active garment recognition and target grasping point detection using deep learning. *Pattern Recognition*, 74:629–641, 2018.

[9] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[10] Thomas Ziegler, Judith Butepage, Michael C Welle, Anastasiia Varava, Tonci Novkovic, and Danica Kragic. Fashion landmark detection and category classification for robotics. In *2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 81–88. IEEE, 2020.

[11] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *European Conference on Computer Vision*, pages 512–530. Springer, 2020.

[12] Li Sun, Gerardo Aragon-Camarasa, Simon Rogers, Rustam Stolkin, and J Paul Siebert. Single-shot clothing category recognition in free-configurations with application to autonomous clothes sorting. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6699–6706. IEEE, 2017.

[13] Li Sun, Simon Rogers, Gerardo Aragon-Camarasa, and J Paul Siebert. Recognising the clothing categories from free-configuration using gaussian-process-based interactive perception. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 2464–2470. IEEE, 2016.

[14] John Schulman, Alex Lee, Jonathan Ho, and Pieter Abbeel. Tracking deformable objects with point clouds. In *2013 IEEE International Conference on Robotics and Automation*, pages 1130–1137. IEEE, 2013.

[15] Andreas Verleysen, Matthijs Biondina, and Francis Wyffels. Video dataset of human demonstrations of folding clothing for robotic folding. *The International Journal of Robotics Research*, 39(9):1031–1036, 2020.

[16] Júlia Borràs, Guillem Alenyà, and Carme Torras. A grasping-centered analysis for cloth manipulation. *IEEE Transactions on Robotics*, 36(3):924–936, 2020.

[17] J. Borràs I. Garcia-Camacho and G. Alenyà. Knowledge representation to enable high-level planning in cloth manipulation tasks. *ICAPS 2022 Workshop on Knowledge Engineering for Planning and Scheduling*, 2022.

[18] Yun Zhou, Shangpeng Ji, Tao Xu, and Zi Wang. Promoting knowledge construction: a model for using virtual reality interaction to enhance learning. *Procedia computer science*, 130:239–246, 2018.

[19] Unity Technologies. [online] unity documentation. https://docs.unity3d.com/Manual/index.html, 2021.

[20] Tom Ward, Andrew Bolt, Nik Hemmings, Simon Carter, Manuel Sanchez, Ricardo Barreira, Seb Noury, Keith Anderson, Jay Lemmon, Jonathan Coe, et al. Using unity to help solve intelligence. *arXiv preprint arXiv:2011.09294*, 2020.

[21] A Juliani, VP Berges, E Vckay, Y Gao, H Henry, M Mattar, and D Lange. Unity: A general platform for intelligent agents. arxiv 2018. *arXiv preprint arXiv:1809.02627*.

[22] Lucas Alberto E Pineda Metz. *An evaluation of unity ML-Agents toolkit for learning boss strategies*. PhD thesis, 2020.

[23] Maryam Honari. Unity-technologies ml-agents. https://github.com/Unity-Technologies/ml-agents, 2013.

[24] Arthur Juliani, Vincent-Pierre Berges, Ervin Teng, Andrew Cohen, Jonathan Harper, Chris Elion, Chris Goy, Yuan Gao, Hunter Henry, Marwan Mattar, et al. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*, 2018.

[25] Virtual Methods Studio. [online] obi documentation. http://obi.virtualmethodstudio.com/, 2022.

[26] Steve Streeting and Rojtberg Pavel. Ogre mesh xml. https://github.com/OGRECave/ogre/blob/master/Tools/XMLConverter/docs/ogremeshxml.dtd, 2018.

[27] Irene Garcia-Camacho, Júlia Borràs, Berk Calli, Adam Norton, and Guillem Alenyà. Household cloth object set: Fostering benchmarking in deformable object manipulation. *IEEE Robotics and Automation Letters*, 7(3):5866–5873, 2022.