Self-reconfigurable Modular Robot Interface Using Virtual Reality: Arrangement of Furniture Made out of Roombots Modules*

Valentin Nigolian[1], Mehmet Mutlu[1,2], Simon Hauser[1], Alexandre Bernardino[2] and Auke Ijspeert[1]

***Abstract*— Self-reconfigurable modular robots (SRMR) offer high flexibility in task space by adopting different morphologies for different tasks. Using the same simple module, complex and more capable morphologies can be built. However, increasing the number of modules increases the degrees of freedom (DOF) of the system. Thus, controlling the system as a whole becomes harder. Indeed, even a 10 DOFs system is difficult to consider and manipulate. Intuitive and easy to use interfaces are needed, particularly when modular robots need to interact with humans. In this study we present an interface to assemble desired structures and placement of such structures, with a focus on the assembly process. Roombots modules, a particular SRMR design, are used for the demonstration of the proposed interface. Two non-conventional input/output devices - a head mounted display and hand tracking system - are added to the system to enhance the user experience. Finally, a user study was conducted to evaluate the interface. The results show that most users enjoyed their experience. However, they were not necessarily convinced by the *gesture control*, most likely for technical reasons.**

***Index Terms*— Human robot interface, modular robots, visual feedback, virtual reality, gesture recognition, gesture control, self-reconfiguration**

I. Introduction and Motivation

A wide variety of robots have specific designs to perform efficiently at a given task. Indeed, designing the robot specifically for the task is more likely to succeed when high precision, accuracy, performance and efficiency-like criteria are required. However, another criterion can be the multi-purpose usage. In other words, a robot may perform many different tasks to adapt to unexpected scenarios and changes in the environment when it has a multi purpose design. There are different levels and ways of achieving generic-use robots. One of the extreme branch for multi-purpose robots is called modular robots (MRs) [1]. The idea behind MRs is having relatively simple modules that are not capable of performing many tasks, but drastically enhance their capabilities when multiple modules work together in a collaborative manner. If the reconfiguration can be done autonomously by an MR, the type of MR is called a Self-Reconfigurable Modular Robot (SRMR). SRMRs can potentially offer solutions to complex problems by using many of simple building blocks, and by controlling them in a coordinated way.

[2] and [3] give extensive literature surveys of MRs in terms of mechanical and control approaches respectively. There is a large literature behind challenges of SRMRs.

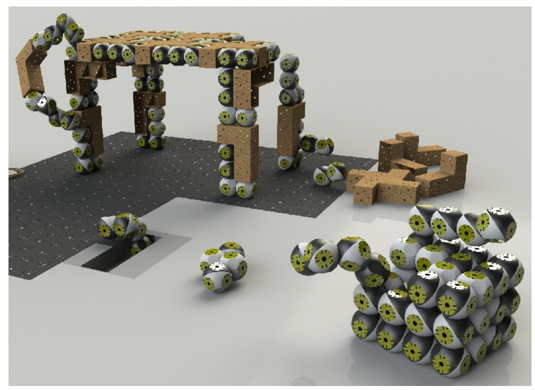


Fig. 1. A single Roombots module consists of four half-sphere-like parts that can rotate independently. There is a continuous rotation capability between each dark and white hemisphere within a module. They can form more complex structures through self-reconfiguration. The rendered image displays different capabilities of Roombots while reconfiguring themselves into a table by utilizing L-shaped and X-shaped passive elements.

Roombots (RB) is type a SRMR designed and developed at the Biorobotics Laboratory (EPFL, Switzerland) [4]. Three main aims of the RB project can be summarized as follows: (i) Studying distributed locomotion control of different body morphologies on unknown terrains when multiple modules come together and form a structure; (ii) exploring self-organization and collaboration algorithms thanks to RBs' self-reconfiguration capability; (iii) creating self-reconfigurable adaptive furniture which also inspired the name "Roombots". Particularly for the last aim, RB needs to interact with humans. A user should be able to design new furnitures and then set desired locations for the needed furniture in the workspace. The structure made out of many RB modules can be seen in Fig. 1. A single module has three inner axes and thus three actuated DOF that can rotate continuously. By using retractable claw-like active connection mechanisms (ACM), modules can attach to each other and to engineered grid surfaces. The single modules are relatively simple, yet they can achieve complex tasks when multiple modules cooperate. Moreover, Roombots modules can be used in conjunction with passive elements of various

*This work was supported by the Fundação para a Ciência e Tecnologia (FCT) agency of Ministry for Education and Science of Portugal under the contract number PD/BD/105781/2014 and project grant AHA-CMUPERI/HCI/0046/2013 and the Swiss National Science Foundation (Project 153299).

[1]V. Nigolian, M. Mutlu, S. Hauser and A. Ijspeert are with Biorobotics Laboratory, School of Engineering and Institute of Bioengineering, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland valentin.nigolian@epfl.ch, mehmet.mutlu@epfl.ch, simon.hauser@epfl.ch, auke.ijspeert@epfl.ch

[2]Mehmet Mutlu and Alexandre Bernardino are with Electrical and Computer Engineering of the Faculty of Engineering at IST, Instituto Superior Téchnico, 1049-001 Lisbon, Portugal alexandre.bernardino@tecnico.ulisboa.pt

shapes such as cross- or L-shapes. They are simple box-like objects allowing to reduce the number of modules required to build a furniture. Their use results in a reduction of costs without impinging on the structural soundness.

Our aim in this work is to create a user interface for assembly of structures made out of RB modules and placement of said structures. Essentially, this work focuses on the problem of assembling structures and its intricacies. The MR community already makes use of visualization tools. A nice example is given in [5] where different structures made out of molecubes are created in a simulation environment. High DOF structures created with modular robots increase the need of user-friendly interfaces. For instance, an average remote controlled model car has two degrees of freedom (steering and throttle) and a user can fully control it. In contrast, a human cannot directly control all individual actuators of a 60-DOF SRMR structure. Such a task requires a higher level abstraction layer for a human operator control. However, no high-level interface to control and assemble SRMRs could be found in the literature. This means that although the various hardware-related aspects of SRMRs have been studied in length, very little work has been done regarding how to interact with them. As it follows, it is very difficult to find inspiration when designing such a new interface and even more difficult to evaluate it, as there is no suitable comparison. For this reason, the underlying basis of the interface we present is mostly empirical.

The contributions of this paper are twofold. First, an interface is proposed to do assembly of structures made out of SRMRs in a seemingly natural manner. The user experience is enhanced by integrating a head mounted display and a hand tracking system to the interface. The interface was then evaluated by conducting a user study. The rest of the paper is organized as follows. Sec. II gives a review of VR techniques used in related concepts. The proposed system is presented in Sec. III and the evaluation of the system is explained in Sec. IV. Finally, concluding remarks and future works are given in Sec. V. Additionally, an explanatory video showcasing the interface's features was made.

II. Background on Virtual Reality

Virtual reality (VR) tools are becoming a part of daily life and they are increasingly being used. Two of the most common uses for VR are video games and online product customization. Changing color, texture or other features of a virtual product and displaying it from any angle is widely used to make the product more appealing. One of the first examples of online product customization is presented in [6]. Customers can design their custom furnitures (shelves), online, within the allowed limits so that the product can be manufactured. In [7], the similarity of immersive virtual environments are compared with real physical mock ups for office related tasks on various architecture, engineering and construction works. Authors conclude that immersive virtual environments can successfully represent the physical world in most of the cases. Substitutionary reality is another branch similar to VR addressing the problem of differences between real and virtual objects around the user [8]. In substitutionary reality, all real objects are associated with virtual counterparts with a certain discrepancy. A low cost VR framework is presented in [9] where authors focus on the framework that they present and emphasize the use of VR in training scenarios.

One of the most used applications for the VR is spatial object placement. Particularly, furniture placement in a room is addressed frequently. The whole mock-up can consist of real-sized virtual objects which the user can interact with as if they are real, as shown in [10] where authors use their system to review the design of a hospital room. Another similar VR application is given in [11] where authors use Oculus Rift and Leap Motion in their interface to help a CAD assembly task. Moreover, virtual environments have the intrinsic feature of modifying physics. This allows for physically-impossible behaviors such as objects floating or intersecting each other.

VR offers useful tools for robot control and visualization. Robots may be far away from the operator and the operator can face a challenging scenario. Indeed, analyzing the situation can be extremely difficult with only raw data. However, an appropriate use of a VR tool can ease the duty of the operator and can increase the mission performance as explained in [12] where VR is used while controlling remote space rovers. Another important aspect of VR in robot control is the ability to have different viewing angles. Although authors are using only simulation in [13], they are analyzing the effects of the view angle in formation control tasks of robot swarms. In the real world, all viewing angles may not be possible, but recreating the robot state in a virtual environment gives more freedom to an operator.

In a previous study on the same RB hardware [14], authors studied an interface with a goal similar to this current work. The approach in [14] involves study on a human robot interface in a real-life-size virtual environment using a tablet PC. They found that being able to move in the virtual environment increases precision in furniture placement. They also concluded that augmented reality (placing virtual objects on the real video captured by the mobile device) does not have significant effect on the same spatial arrangement task. Another modular robotics interface presented a direct human robot interaction [15]. In that study, the operator only points towards a RB module using an arm to chose it and afterwards the operator can point a desired goal location for the chosen module. Then, the module autonomously locomotes to the goal location on the grid environment. The pointing gesture is tracked by a Kinect. This natural interface considers only replacement of existing single modules and it does not support self-reconfiguration from one structure to a new one. If the notion of placement using VR has been extensively studied, this knowledge can be adapted to manipulate modules in a way that they are assembled to create new structures. This would be an interface which would consider modules to be brick-like objects, with the user assembling structures brick-by-brick. However, it would not consider the ability of SRMRs to change their configuration. The interface we present in this work is intrinsically related to the very essence of SRMRs.

III. System Design

RB are designed to work in a shared environment with humans. In order to control many RB modules, there is a need for practical and easy-to-use interface. The demonstrative scenario considered in this work is building arbitrary structures with RB modules, particularly furniture.

A. Requirements of the System

The primary consideration of this work is usability. We envision a scenario where the user has many RB modules and those modules can create furniture according to needs. For instance, a table made out of RB modules should be able to self reconfigure into chairs and/or stools when needed, thus saving space and having redundant modularity at home/work. The user interface should provide a simple way to define those two different configurations with a pre-established set of modules and passive elements. VR enables also the remote control, such that the user can change the furniture configuration even if he/she is not present in the room. To sum up;

- The user should be able to create any desired structure, using RB modules.
- The user should interact with the furniture in a virtual environment which is the representative of the real room.
- The environment should support innovative input devices to provide an intuitive interface.
- The system is expected to make the user feel immersed in the virtual environment.

B. Overall System

The interaction is designed to be completely in a virtual environment in order to satisfy the design requirements of interacting with the RB furniture. The system consists of a virtual environment representing a workshop and a room and a Graphic User Interface (GUI) to interact with said environment. The minimal requirements of the interface is only a regular PC filling the system requirements of the Oculus Rift. In our system, the computer uses Microsoft© Windows 7 as an OS and acts as the central device binding the other elements together.

There is a substantial number of different areas of interaction which can potentially result in meeting our design criteria. Joystick, gamepad, inertial remote controllers, haptic devices projectors and 3D screens are only some of the most popular input-output devices that could have been used. Since the system represents the real world, devices with more real-life-like interaction would be more appropriate. Hence, two extra devices are chosen to enhance the user experience: Oculus Rift, a VR device consisting of a head mounted display for 3D vision with head tracking capability, and Leap Motion, a hand tracking and gesture recognition (GR) system. Leap Motion introduces the concept of holding items with the hand and moving them around. On the other hand, Oculus Rift not only replaces the screen with 3D vision capability, but also gives head pose tracking capability to steer the direction of sight in a life-like way.

1) Environment Components: The environment contains various elements interacting together. We distinguish four of them :

- Roombots modules, as described in Sec. I
- L-shapes, passive elements on which Roombots modules can attach. They can be seen on Fig. 1.
- Passive plates, simple static elements on which Roombots modules can attach. They are typically fixated on walls, the ceiling or the floor. Such plates can also be seen on Fig. 1 as the dark gray area on the floor.
- Structures, a set of Roombots modules and L-shapes assembled together.

2) Virtual Environment: The virtual environment of this system is a representation of the real environment where one would assemble modules together and a room which can be any place where furniture is needed; a house, an office, a hospital, a school, a garden and so on. In our specific implementation, an indoor place is considered. This environment has two aspects : the *Workshop* and the *Room*. In the Workshop, Roombots modules and their passive companions can be manipulated to create arbitrary structures. It features a large rotating plate that can be used to consider one's current work from various angles and a grid of passive connectors on which modules can be attached. After building a structure, the user can "export" it to the "room". In the Room, the custom structures are considered to be a single object and are manipulated as such. The user can place the structures in the room along with pre-defined furnitures such as a stool or a table. On the visual side, the Room is not very refined, as this work is more of proof-of-concept than a finite product. We consider two rooms and not one to clearly distinguish the two sort of tasks they allow to do. The Workshop aims at *building* structures. Modules and passive elements are considered as distinct objects. On the other hand, the Room aims at *placing* structures as they would be in an actual room. Whole sets of modules and passive elements are considered as single objects. Screenshots from the workshop and the room can be seen in Fig. 2 and Fig. 3 respectively.

3) Graphical User Interface (GUI): The GUI has four main components: the holders, the pointer, the turntable and the display. First, the holder contain the Roombots modules and L-shapes. They provide a way to add new elements to the scene, as new ones will appear once an existing object has been displaced. Note that the holders are purely conceptual and are thus not shown. The second component is the pointer. It allows to pick up the structures from the holders or anywhere else. The 3D motion of this pointer can be controlled by the Leap Motion. Thirdly, the "turntable" is a large plate that can be rotated, thus giving different perspectives on the modules that are attached to it. Finally, the display shows the current state of the environment through the Oculus Rift's HMD. The Oculus Rift also acts as an input device since its orientation is interpreted by the software to define which part of the scene must be displayed.

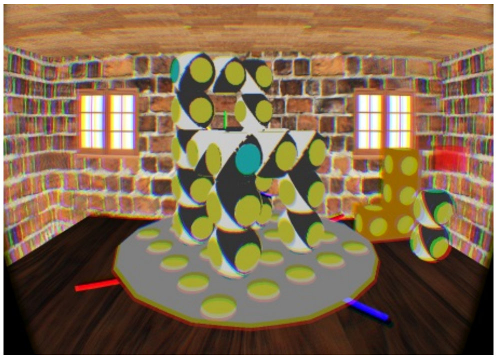


Fig. 2. Main components of the GUI in the Workshop. On the right, RB modules and L-shapes can be grabbed and manipulated (the red cube represents the hand-controlled pointer). The cube turns white when holding a handle, to give the user feed-back that he or she pinched it. At the bottom, the "turntable" and its handles. In the center, the structure being currently assembled.

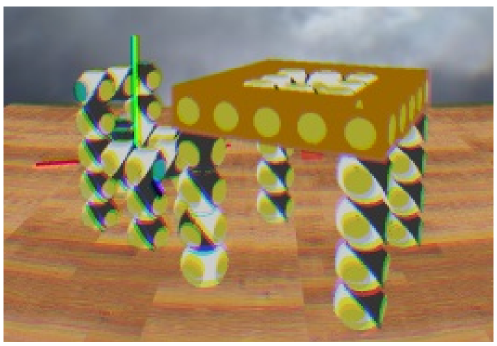


Fig. 3. Users can place their custom structures anywhere in the Room. Here, we can see the same chair as in fig.2, placed in front of a pre-built table made of Roombots modules and L-shapes.

Indeed, the core functionality of any VR device is to follow the user's head movements. Furthermore, the interface uses the keyboard to perform generic tasks that are independent from the other devices. It allows to move around in the room using keys in an arrow-like configuration on the left-side of the keyboard, import and export structures and switch between the Workshop and the Room.

C. Giving More Immersive Feeling

Using Oculus Rift and Leap Motion devices, we aimed to give a more immersive feeling.

1) Directly grasping objects: Leap Motion, [16], is used to track hand gestures. The same function can also be achieved by other depth image sensors like Kinect. Leap Motion was chosen assuming that it can perform better since it is specifically designed to track hand gestures. Holding objects with e.g. a mouse has only two states, holding or not-holding depending on mouse button state and its real world 2D motion must be transformed in a virtual 3D one which is a common problem in 3D-based softwares. However, Leap Motion provides 3D motion and a continuous state for the pinching (grabbing) gesture depending on the closeness of fingers. The GUI gives a holding state feedback by altering the size of pointer. In other words, open hand (non-holding state) results in a big pointer and grabbing hand (holding state) displays a small pointer. Additionally, the inner cube turns white when the user is dragging an object's "handles". Handles are presented in Sec. III-C.2. Thus, the user can get feedback from the system if the grabbing attempt is not successful, e.g. when the gesture is not done correctly. Fig. 4 illustrates the operator while using the complete set-up to assemble RB-made structures and place them. Indeed, grabbing structures is done by pinching with the hand while bringing the pointer to an object.

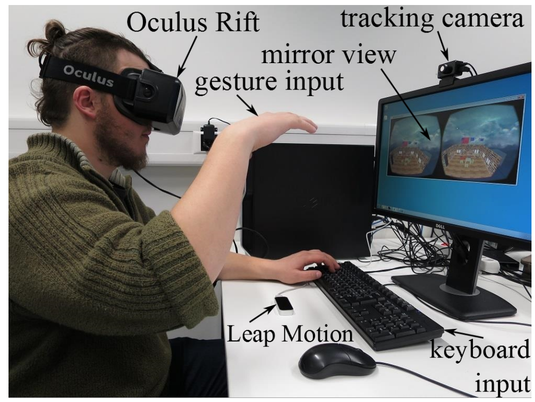


Fig. 4. The complete set-up required by the system during a user test.

2) Object Manipulation & Handles: Ideally, one would want to use the same gestures as in real life to manipulate an object. To rotate a module's components, both hands are needed, one holding the module and the other rotating another part of it. However, recognition of complex hand gestures is quite a difficult task and no current technology allows to perfectly mirror hands. For this reason and to keep the interface as simple as possible we only use one gesture, namely the "pinch". The idea is to use "handles", analogous to rods attached to the objects and their components. When an object has been grabbed, it is considered to be "selected". Once selected, an object shows handles that are used to manipulate it, rotation-wise. Pinching a handle and then dragging it would have the object rotate such that the handle follows the direction to the user's hand. Three handles are used. One follows exactly the hand's movements and the other two makes the object rotate around the first one. For a Roombots module, three more handles are added to change its configuration by rotating its components around their axis. All three correspond to rotation around its three aforementioned inner axes, i.e. the actuated degrees of freedom of the real modules. See Fig.5 for a visual explanation of handles.

D. Assembly

The "connector" is the main concept underlying the assembly process. All objects possess a set of connectors that are either passive or active and that are used to connect them together. Active connectors correspond to the actual ACMs present in all Roombots modules. They represent the ability to attach to other modules' passive connectors. Two objects are connected if a connector of the first is connected

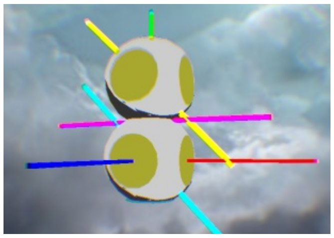


Fig. 5. A Roombots module in current selection, thus showing its handles. The green handle follows exactly the direction to the hands position while the red and blue handles only rotate around the green handle. The rotation of the green handle is done around the bottom of the module. The cyan, magenta and yellow handles are used to change the Roombots module's configuration.

to a connector of the second. More than manipulate single objects using handles, the interface aims at assembling various objects together. This task has two main components : snapping and connectivity. Snapping is changing an object orientation so their connectors match perfectly when they are brought close. It is a well-known rigid body problem and its implementation will not be explained further.

1) Connectivity: The connectivity problem, however, is non-trivial. Defining how the connections between the objects are represented has several aspects. In this interface, a hierarchical model is used. When two objects are connected, one is considered to be the "parent" and the other the "child". Manipulating the parent (e.g. dragging it) results in forwarding the modification to its children in a recursive manner, thus manipulating bulks of objects as a whole. To deconstruct a bulk, one would have to grab the latest-attached modules. While this approach works well when assembling a structure piece-by-piece, it becomes more complicated when connecting bulks of objects. Indeed, when connecting two sets of objects together, each having their own "parent", it is unclear which of them should become the parent of the connected structure. All relationships would have to be redefined in any case. No convenient solution was found to this problem. When two bulks are connected, modifying one will thus not forward the modification to the other. See Fig. 6 for a more detailed explanation of the problem.

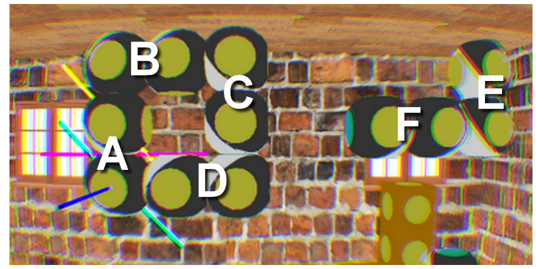
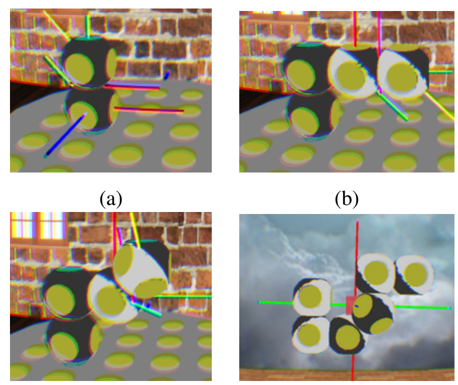


Fig. 6. On the left, a module loop forming a bulk. Module A is parent of B, which is parent of C which itself is parent of D. However, the chain stops here to avoid any infinite-update loop. D is thus parent of no module. Moving A would move the whole bulk, but moving C would only move D in addition. The same goes for any modification of configuration. On the right, a second bulk with E being the parent of F, hence the parent of this second bulk. If the user were to bring the left-hand bulk close-enough to F, the whole bulk would be snapped, but no connection would be made (even though the connector of F is active, as shown by its pale blue color). Indeed, after this connection, is is unclear which module becomes the parent of all modules.



(a) (b)

(c) (d)

Fig. 7. Operators were asked to perform four successive tasks. First (a), simply connect a module to the turntable without rotating any of the joints. Second (b), rotate a module using two handles and connect it to the first one. Third (c), change the second module's configuration to match the picture. Finally (d), add a third module on top, switch to the "Room" and rotate the whole structure to match a given picture.

IV. Results and Discussions

A user study was conducted to test the interface and see if it is a feasible and user-friendly approach. 21 volunteered test subjects (operators) with various VR and GR background tried the system. The main focus was making them perform a series of increasingly difficult construction tasks, thus testing the main functionalities of the interface. Subjective system appreciation was used to evaluate the proposed interface. They simply used the interface as it is. Fig. 7 gives more detail on the task performed by participants.

A. User Study

Once the users completed their tasks, they were asked the questions in Table I. Possible choices were: (0) not at all, (1) not really / no, (2) a little bit / neutral, (3) yes and (4) a lot / definitely depending on the question. Additionally, they were asked the following background questions: "Did you have any previous knowledge about VR?", "Did you have any previous knowledge about GR?" and "Did you have any experience in 3D environments (Games, Computer-Aided Design, etc.)?". Those questions were used to determine the prior experience of our users regarding methods used by the interface. Answers to those background question yield that our sample had sparse (high standard deviation) and overall balanced (mean close to "neutral") experience of both VR and GR, with GR being less known than VR.

The results yield an overall positive feedback, with most people enjoying their experience. This is shown by Q7, with a mean well above "yes" and little variation ($\sigma = 0.51$). In contrast, participants did not find the gesture-control particularly intuitive, as shown by Q2. However, there seems to be consensus that it becomes easier to use with time (learning effect). Indeed, answers to Q3 indicates that participants felt more comfortable after the initial adaptation

TABLE I

SUBJECTIVE EVALUATION OF THE PROPOSED INTERFACE (RANGE: 0-4)

	Question	μ	σ
Q1	Did the OR help you perceive the depth better?	2.71	0.85
Q2	Is the gesture control intuitive?	2.86	0.79
Q3	After some time, did you feel the interface getting easier to use?	3.33	0.58
Q4	Did you feel immersed with the OR?	3.57	0.60
Q5	Did you feel immersed with the LM?	2.76	0.60
Q6	Overall, how would you rate this interface?	2.83	0.56
Q7	Overall, did you enjoy your experience?	3.52	0.51
Q8	Would you use this interface if you had to do a similar assembly job?	2.38	1.16
Q9	Would you use another VR-based and GR-based interface to do a similar job?	2.76	0.94

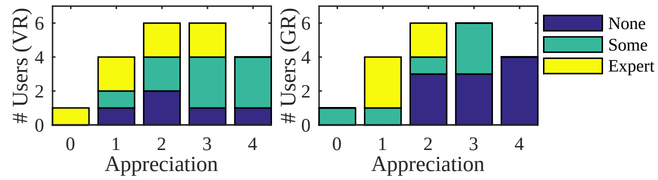


Fig. 8. The subjective opinions of users are asked to evaluate appreciation (Q8) distribution of users with respect to their prior experience for the proposed interface. Two plots consider VR and GR experience respectively. Three experience levels consist of (i) none or almost none, (ii) limited and (iii) high experience. Experienced VR users are hard to satisfy since they already had long enough interaction with commercial applications, whereas less experienced VR users appreciated our interface more. A similar trend can also be seen in GR experience levels. However, our users had less GR experience compared to their VR experience and they reacted with more enthusiasm.

time. This concurs with the examiner's observations that after a short period of confusion, mostly regarding reconfiguration, most subjects tended to rapidly gain understanding of how to use them. It must be noted that even though people *felt* they got better, a few of them still had great difficulties in mastering the proposed gesture control. Additionally, no participant was familiar with the notion of SRMRs. Q4 and Q5 strongly imply that subjects felt immersed using a VR and GR-based interface, with the Oculus Rift having very high consensual approval and the Leap Motion lower but still satisfactory approval. Q8 and Q9 show an overall positive feed-back but also distinct discrepancies regarding whether or not participants would use this interface or a similar one. Indeed, subjects were not necessarily convinced by the concept of gesture-control itself and a few of them argued they would be more efficient using a more traditional approach. For this reason, the data of those two questions are further discussed using two graphs in Sec. IV-B.

B. Discussions

To sum up the user study, we asked the users to perform simple tasks using the immersive interaction framework. The results are promising, yet, should be further discussed in details. Indeed, Q2 and Q8 yield disappointing results. Concerning Q2, the main problem is, as far as we could observe, the detection of the hand due to technical limitations. For example the grabbing/pinching gesture works better when the gesture angle is similar to holding a vertical stick than a horizontal stick. In addition, the range of the Leap Motion is spatially quite limited and gets less and less precise as the hand reaches its boundary. As a result, participants were sometimes confused by why it did not work when stretching their arm too far from the sensor. This is consistent with the results of [11], and should deserve more attention when trying to design a similar interface.

The solution offered by the interface, to let the user move in the room using the keyboard, compensates for this limitation but was observed to often be counter-intuitive. It thus requires training to learn that unexpected sensor response and our test subjects did not have enough time to fully get used to this response. It is interesting to observe that while people would use a similar interface to perform a similar assembly task, they were not particularly convinced by the one presented in the present work.

Additionally, if it did not convince subjects with no or no real experience of VR or, on the contrary, a subject with broader knowledge of it, people possessing little to moderate VR background were more convinced, as shown in Fig.8. This might mean that people with little VR background would be more pleased by our use of this technology than people with no real or no background. The latter might not see the benefits of VR to do such tasks. A similar observation can be made regarding Gesture Recognition. As shown in Fig. 8, the less experienced participants were, the more they were willing to use the interface again. Arguably, people with GR experience might have done so in a completely different manner and were thus forced to change their habits.

However, with little practice, one can be quite efficient at assembling structures. Indeed, the developer of this interface is able to build arbitrarily complex structures in a very natural way. Fig. 9 shows an example of a complex structure built by an experienced user. Finally, the results of Q7 are are positive. Which shows that even if subjects would not necessarily use such an interface to perform similar assembly tasks, they still enjoyed their experience. Indeed, VR and GR make the interfaces using those concepts more appealing.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a novel self-reconfigurable modular robot interface to be able to command Roombots to create desired furniture on the needed spot. The interface uses a virtual environment to represent the real world and has two different functionalities: Create structures using the Workshop and then place them using the living space. To give a more immersed control to the user, HMD and hand tracker were adopted to the interface. Although our user study is not statistically very significant due to the number of testers, we conclude that while there is still room for improvement, especially regarding to gesture-recognition, this interface and similar approaches are promising and open to novel ways to interact with robots in general and even more so with

Fig. 9. This throne shows the possibilities of Roombots modules. It uses modules in various configurations and positions. All physically possible structures can be built by an experienced user, no matter how complex it may be.

SRMRs. The “handles” approach appears to be a convincing way to achieve intuitive control over the modules considering the low complexity of gesture recognition solutions opted for (i.e. pinching). The handles have certain drawbacks (clarity, not suited for color-blindness, crossing depending on configuration, etc.) but provide a way to create any structure in a much faster way than manually setting the rotary joint angles of each module.

It is important to note that we do not claim that the presented interface accelerates the task completion or increases precision of user control compared to any other. It is an *attempt* to give the user more immersive feeling while controlling SRMR. We evaluated its qualitative *accessibility* and not its quantitative *performance*.

The study’s results clearly show the interest of participants for VR and GR. This ”fun factor” is a well-known phenomenon related to VR, GR and gamification. Gamification is the subject of many studies and has a broad set of possible applications, notably teaching, as shown by [17]. Incidentally, such a strong appreciation for our interface from participants implies it could be used not only to build structures but also to *learn how to* build such structures and manipulate SRMRs. Indeed, SRMRs can be quite complex and it takes time to learn how they behave since their reconfiguration might not be intuitive (efficiency and capability are the priority over practicality). Being able to ”play” or ”fiddle” with them virtually is very helpful to understand their mechanisms. As such, our interface might provide a more natural way to study SRMRs.

Currently, the main draw-back of the interface is that it is limited in gesture recognition. While the “pinch” is very simple to detect and intuitive, the interface would reach a different level if the hands could be used in a completely natural manner. A possibility to attain this level might be to use haptic gloves or similar gesture-recognition devices. SRMRs still present great challenges on the reconfiguration side. Indeed, it is very difficult to automatically build a target structure. The present interface could be improved to store data from the manual assembling sequence, data that could be used by the modules to self-reconfigure.

VI. Acknowledgments

Authors would like to thank the user study participants.

References

[1] M. Yim, W.-M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. Chirikjian, “Modular Self-Reconfigurable Robot Systems [Grand Challenges of Robotics],” *IEEE Robotics Automation Magazine*, vol. 14, no. 1, pp. 43–52, Mar. 2007.

[2] H. Ahmadzadeh, E. Masehian, and M. Asadpour, “Modular Robotic Systems: Characteristics and Applications,” *Journal of Intelligent & Robotic Systems*, pp. 1–41, Jun. 2015.

[3] H. Ahmadzadeh and E. Masehian, “Modular robotic systems: Methods and algorithms for abstraction, planning, control, and synchronization,” *Artificial Intelligence*, vol. 223, pp. 27–64, Jun. 2015.

[4] A. Sprwitz, R. Moeckel, M. Vespignani, S. Bonardi, and A. J. Ijspeert, “Roombots: A hardware perspective on 3d self-reconfiguration and locomotion with a homogeneous modular robot,” *Robotics and Autonomous Systems*, vol. 62, no. 7, pp. 1016–1033, Jul. 2014.

[5] V. Zykov, A. Chan, and H. Lipson, “Molecubes: An open-source modular robotic kit,” in *in IROS-2007 Self-Reconfigurable Robotics Workshop*, 2007.

[6] M. Nousch and B. Jung, “CAD on the World Wide Web: Virtual Assembly of Furniture with BEAVER,” in *Proceedings of the Fourth Symposium on Virtual Reality Modeling Language*, ser. VRML ’99. New York, NY, USA: ACM, 1999, pp. 113–119.

[7] A. Heydarian, J. P. Carneiro, D. Gerber, B. Becerik-Gerber, T. Hayes, and W. Wood, “Immersive virtual environments versus physical built environments: A benchmarking study for building design and user-built environment explorations,” *Automation in Construction*, vol. 54, pp. 116–126, Jun. 2015.

[8] A. L. Simeone, E. Velloso, and H. Gellersen, “Substitutional Reality: Using the Physical Environment to Design Virtual Reality Experiences,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI ’15. New York, NY, USA: ACM, 2015, pp. 3307–3316.

[9] T. Hilfert and M. Knig, “Low-cost virtual reality environment for engineering and construction,” *Visualization in Engineering*, vol. 4, no. 1, pp. 1–18, Jan. 2016.

[10] P. S. Dunston, L. L. Arns, J. D. Mcglothlin, G. C. Lasker, and A. G. Kushner, “An Immersive Virtual Reality Mock-Up for Design Review of Hospital Patient Rooms,” in *Collaborative Design in Virtual Environments*, ser. Intelligent Systems, Control and Automation: Science and Engineering, X. Wang and J. J.-H. Tsai, Eds. Springer Netherlands, 2011, no. 48, pp. 167–176.

[11] N. Beattie, B. Horan, and S. McKenzie, “Taking the LEAP with the Oculus HMD and CAD - Plucking at thin Air?” *Procedia Technology*, vol. 20, pp. 149–154, 2015.

[12] L. A. Nguyen, M. Bualat, L. J. Edwards, L. Flueckiger, C. Neveu, K. Schwehr, M. D. Wagner, and E. Zbinden, “Virtual Reality Interfaces for Visualization and Control of Remote Vehicles,” *Autonomous Robots*, vol. 11, no. 1, pp. 59–68, Jul. 2001.

[13] C. T. Recchiuto, A. Sgorbissa, and R. Zaccaria, “Usability evaluation with different viewpoints of a Human-Swarm interface for UAVs control in formation,” in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, Aug. 2015, pp. 523–528.

[14] S. Bonardi, J. Blatter, J. Fink, R. Moeckel, P. Jermann, P. Dillenbourg, and A. J. Ijspeert, “Design and evaluation of a graphical iPad application for arranging adaptive furniture,” in *2012 IEEE RO-MAN*, Sep. 2012, pp. 290–297.

[15] A. Özgur, S. Bonardi, M. Vespignani, R. Möckel, and A. J. Ijspeert, “Natural user interface for Roombots,” in *2014 RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, Aug. 2014, pp. 12–17.

[16] “Leap Motion.” [Online]. Available: www.leapmotion.com

[17] S. Villagrasa, D. Fonseca, and J. Durán, “Teaching case: applying gamification techniques and virtual reality for learning building engineering 3d arts,” in *Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality*. ACM, 2014, pp. 171–177.