

## Article

# Interactive Parametric Design and Robotic Fabrication within Mixed Reality Environment

Yusuf Buyruk \*  and Gülen Çağdaş 

Architectural Design Computing Graduate Program, Department of Informatics, Graduate School,  
Istanbul Technical University, Istanbul 34367, Turkey

\* Correspondence: buyruk17@itu.edu.tr

**Abstract:** In this study, a method, in which parametric design and robotic fabrication are combined into one unified framework, and integrated within a mixed reality environment, where designers can interact with design and fabrication alternatives, and manage this process in collaboration with other designers, is proposed. To achieve this goal, the digital twin of both design and robotic fabrication steps was created within a mixed-reality environment. The proposed method was tested on a design product, which was defined with the shape-grammar method using parametric-modeling tools. In this framework, designers can interact with both design and robotic-fabrication parameters, and subsequent steps are generated instantly. Robotic fabrication can continue uninterrupted with human–robot collaboration. This study contributes to improving design and fabrication possibilities such as mass-customization, and shortens the process from design to production. The user experience and augmented spatial feedback provided by mixed reality are richer than the interaction with the computer screen. Since the whole process from parametric design to robotic fabrication can be controlled by parameters with hand gestures, the perception of reality is richer. The digital twin of parametric design and robotic fabrication is superimposed as holographic content by adding it on top of real-world images. Designers can interact with both design and fabrication processes both physically and virtually and can collaborate with other designers.

**Keywords:** parametric design; robotic fabrication; digital fabrication; mixed reality; augmented reality; digital twin; interactive design; shape grammar; design-to-production; human–robot collaboration



**Citation:** Buyruk, Y.; Çağdaş, G. Interactive Parametric Design and Robotic Fabrication within Mixed Reality Environment. *Appl. Sci.* **2022**, *12*, 12797. <https://doi.org/10.3390/app122412797>

Academic Editor: Alessandro Gasparetto

Received: 30 October 2022

Accepted: 25 November 2022

Published: 13 December 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the use of computer technology, designers have taken their imagination to the next level thanks to the advantages of digital possibilities and they have increased their pursuit of form finding, and started to fabricate forms in complex shapes using new design and production possibilities. However, the production processes of complex design products also contain complex problems. Therefore, the use of computer technologies is not limited to the design phase, but also in the fabrication processes of complex design products.

Parametric-design tools have evolved in such a way that both the design process and the fabrication process can be controlled with parameters. By changing the parameters, the design product as well as the production codes required for the production of these parts are updated. With the development of parametric-design tools, the design model has multiple design alternatives that can be generated with different parameters. The designer reaches a set of results with parametric design instead of a single result, and the possibilities for testing the different alternatives of design products before fabrication are improved. With parametric-design tools, many different design alternatives can be explored by simply changing parameters. The ability to make changes to the design with the parameters has enabled alternatives to be tested and manufactured. In this way, it becomes possible to design and fabricate complex design products.

In this paper, we propose an interactive, parametric design and robotic fabrication method that allows users to dynamically explore design and fabrication alternatives within a mixed-reality environment throughout the whole design and fabrication process. With the proposed method, both parametric modeling and robotic fabrication steps can be created within the mixed-reality environment and controlled by parameters. In order to test the proposed method, a natural stone robotic fabrication environment is created. The proposed method was tested on a design product, which was defined by the shape-grammar method using parametric-modeling tools. Natural stone material was chosen to test the proposed method in robotic fabrication. The results of the proposed method and the existing methods are compared and discussed based on the observations obtained from the test results in terms of mass-customization, design-to-production process, scalability, machine time, process, and material efficiency, and human–robot collaboration. In addition to the production possibilities, design possibilities such as production-immanent modeling, interactive design, emergent design, parametric design, and generative design are offered to the user within the mixed-reality environment.

## 2. Background and Related Work

With the development of new methods and techniques in digital fabrication, industrial robots are more widely preferred in digital fabrication applications. Studies in robotic fabrication have shown that even some hand skills such as stonemasonry and woodcarving can be performed with industrial robots. Carving work on stone surfaces with industrial robots [1] and woodcarving with industrial robots [2] are examples of these studies.

Some studies in robotic fabrication have shown that industrial robots can be used in digital fabrication applications with human–robot collaboration. A metal-assembly study [3] and a timber-assembly study [4] in which users and industrial robots work collaboratively in the same production environment can be given as examples of these studies. In addition, users and industrial robots can work with human–robot collaboration in digital fabrication applications even if they are in different locations [5].

The use of mixed-reality devices in digital-fabrication methods has become increasingly common in recent years. Mixed-reality devices were used in digital fabrication applications such as knitting with bamboo material [6], brick wall assembly [7,8], knitting with metal bars [9], timber structures assembly [4], making a vault structure with Styrofoam pieces [10], and rubble bridge-making [11], as well as in additive manufacturing [12]. Mixed-reality devices were also used in the design and digital fabrication study with composite parts that are stretched and shaped [13].

There are also studies where mixed-reality tools and industrial robots were used together in robotic-fabrication applications. In a robotic wire-cutting-application study, the Styrofoam pieces were produced using an industrial robot and they were knitted using the mixed-reality device [14]. In a study of knitting wooden sticks, an industrial robot was used to notch the joints of wooden sticks, and the mixed-reality device was used during the knitting of wooden sticks [15].

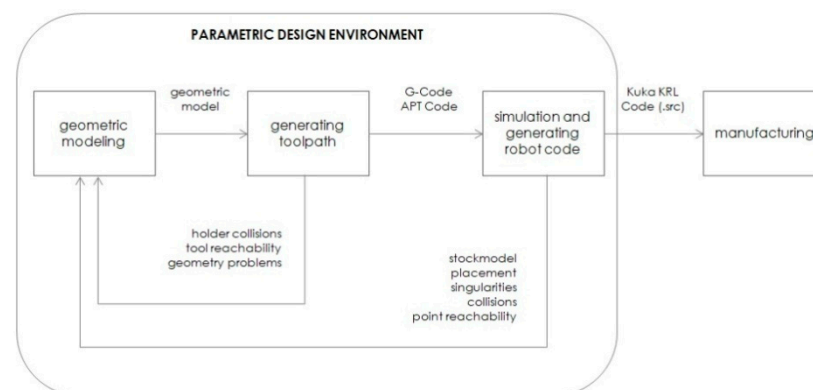
In some studies, mixed-reality devices and industrial robots were used together in design and fabrication processes with human–robot collaboration. In an additive manufacturing study, the industrial robot was used as a 3D printer and the mixed-reality device was used during the design and fabrication steps with human–robot collaboration [16]. In a wire-cutting study with Styrofoam material, the industrial robot and mixed-reality device were used together in the design and fabrication steps, with human–robot collaboration [17]. There are other studies [18–20] in which mixed-reality devices and industrial robots were used together with human–robot collaboration. There are also studies where challenges and opportunities in AR and VR technologies for manufacturing systems [21], and challenges and opportunities in human–robot collaboration are reviewed [22].

### 2.1. Industrial Robot Offline Programming Workflow

Using industrial robots in digital fabrication is called robotic fabrication. In robotic-fabrication applications, the industrial robot offline programming workflow consists of four steps; modeling, toolpath generation, post-process with simulation, and fabrication [23]. In the modeling step, a geometric model is created with computer-aided-modeling (CAD) tools. The toolpath-generation step calculates the path that the cutting tools will follow during the manufacturing of the model with CNC (computer-numerical-control) machines. Computer-aided-manufacturing (CAM) software tools are used in the toolpath-generation step. The generated toolpath code is generally in G-code or APT-code (automatically-programmed-tool) format. The toolpath generated for CNC machines must be post-processed into robot code to be used with industrial robots. While post-processing the toolpath and creating the robot code, it is important to determine the collision risks that the industrial robot may encounter during the fabrication process, to detect errors such as accessibility, exceeding axis limits, and singularity, and to avoid collisions and errors. For these reasons, it is necessary to simulate the industrial robot and its environment before production. These tasks are done in the simulation and robot-code-generation steps. The last step of the robotic-fabrication process is loading the robot code to the industrial robot and running the robot code. This offline programming workflow is linear. Users should follow these four steps in order. If the user wants to make changes in any of the previous steps, the user must repeat other steps that follow. The user can only move on to the next step after completing the previous step.

### 2.2. Parametric Robot-Control Tools

Another method of using industrial robots in robotic-fabrication applications is to create industrial robot programs with parametric-modeling tools. Kuka | PRC [23,24] and ABB HAL [25] plug-ins developed for the Grasshopper3D parametric-design program can be given as examples of this method. With parametric robot-control tools, users can complete modeling, toolpath generating, simulation, and robot code post-processing tasks with parametric-modeling tools. In this way, if any change is made in any of the previous steps, the following steps are automatically updated instantly, and users do not need to repeat other steps that follow. Design-to-production workflow can be managed more flexibly and users can change any desired step with parameters. That both design and robotic fabrication can be controlled by parameters has enabled mass customization [23]. Figure 1 shows the robotic fabrication workflow in the parametric-design environment.

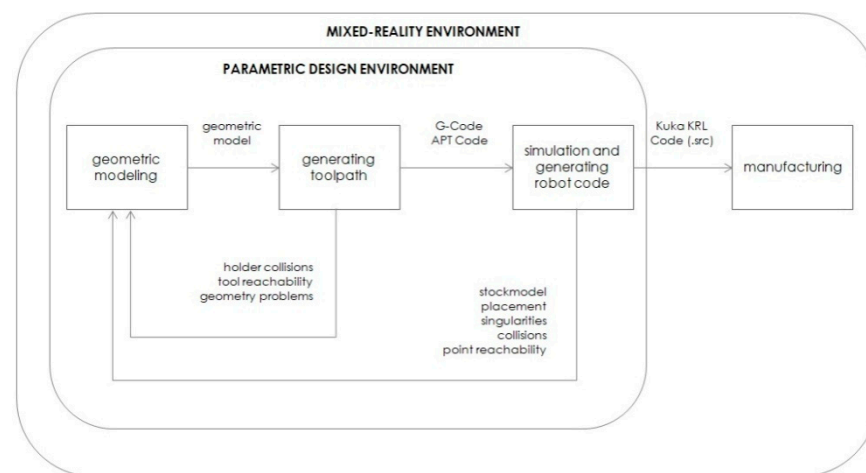


**Figure 1.** Robotic fabrication workflow in the parametric-design environment [23].

## 3. Materials and Methods

In this study, a method for creating parametric design and robotic fabrication steps in a mixed-reality environment is proposed. Users can control parametric design and robotic-fabrication processes with parameters in the mixed-reality environment. Users can also interact physically and virtually with the design and fabrication environment and

make changes at the time of design and fabrication and all the following steps are updated without the need for user intervention. Users can get real-time design and production feedback in the mixed-reality environment. The robotic-fabrication process can continue with human–robot collaboration. In this way, the whole process from geometric modeling to robotic fabrication can be controlled by hand gestures. Simulation images can be viewed as holographic content by adding on the images of the real production environment. Multiple users can coexist in the same holographic environment at the same time and multiple users can interact with the holographic contents in the same parametric design and robotic-fabrication process. In Figure 2, parametric design and robotic fabrication within a mixed-reality-environment workflow can be seen.

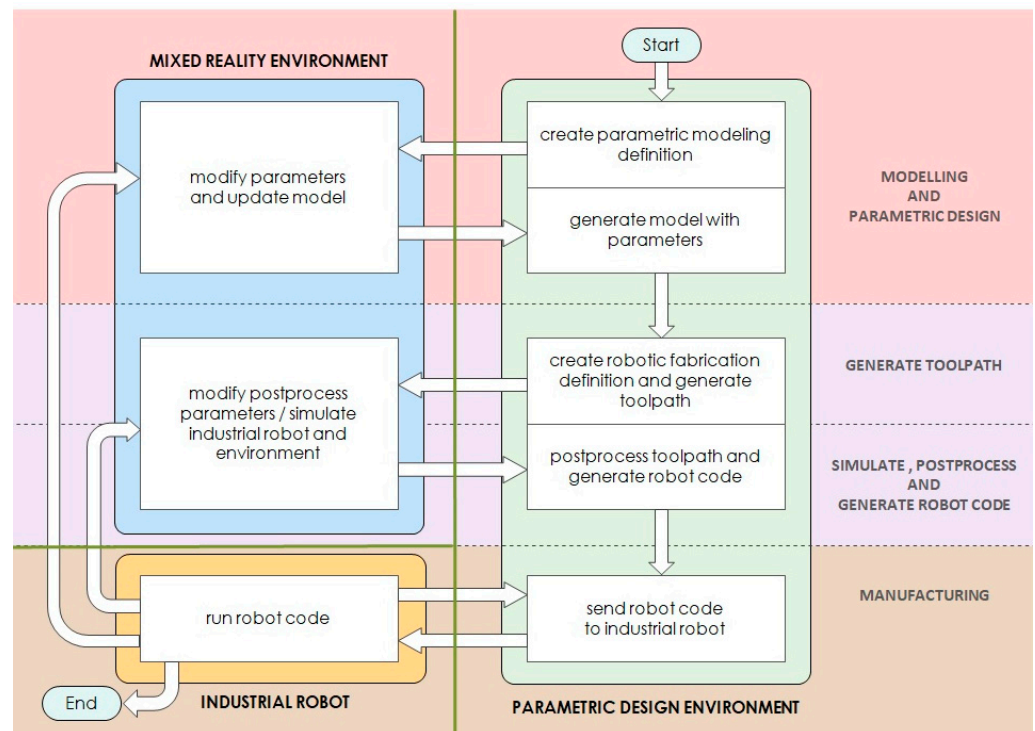


**Figure 2.** Parametric design and robotic fabrication within mixed-reality-environment workflow.

The second generation HoloLens mixed-reality device was used in the study. In the HoloLens mixed-reality device, the holographic content is superimposed on top of the real-world images. The mixed-reality device creates holograms of light and sound objects that look like real objects around us. Holograms can respond to the user’s gaze, gestures, and voice commands. Holograms are created in a holographic virtual world and on the lens in front of the wearer’s eye. The hologram disappears when the angle of view is changed, but if the perspective is directed back to the scene where the object is located, the hologram is displayed again in its real-world location. Users can interact with both real-world objects and the holographic contents in real-time. The mixed-reality device recognizes the boundaries of the real-world environment with its sensors and updates the holographic contents with these boundaries. The mixed-reality device can detect the positions of objects in the real-world, which makes reality perception richer. That user can control holographic content by hand gestures in the mixed-reality environment, strengthens reality perception. In addition, mixed-reality devices allow multiple users to share the same holographic environment and multiple users can interact with the same holographic contents at the same time [26].

In Figure 3, the roles of the mixed-reality tool, the industrial robot, and the parametric-design software in the proposed workflow can be seen.

The initial step of the proposed method is to create the parametric-model definition. Grasshopper3d 1.0 software was used as the parametric-design tool in this study. Grasshopper3d parametric-modeling tool runs inside Rhino3d 7.2 modeling software as a plugin. After the model is defined in the parametric-design program, the user can make changes to the parameters of the model in the mixed-reality environment. The user can monitor the changes in the model in the mixed-reality environment while modifying the parameters of the model.



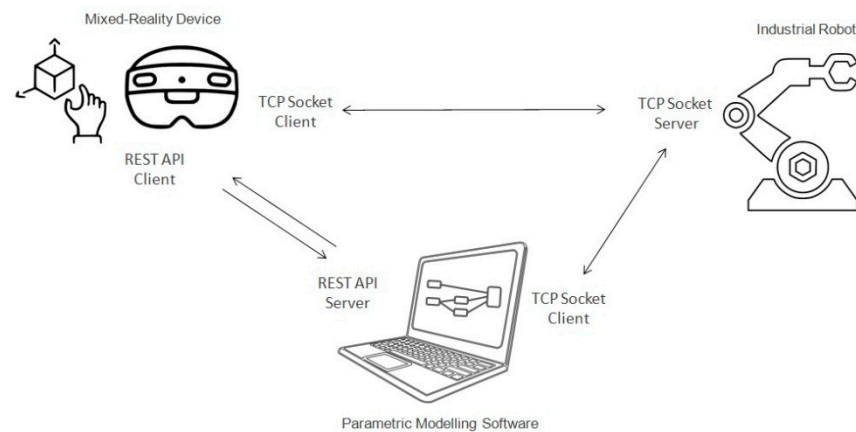
**Figure 3.** The role of the mixed-reality tool, the industrial robot, and the parametric-design software in the proposed workflow.

After the parametric modeling step, the toolpath that will be used to manufacture the model is calculated with the parametric-modeling tool. The generated toolpath must be post-processed and transformed into robot code in order to be used with the industrial robot. At this point, it is necessary to determine the collision risks that the industrial robot may encounter during production, to detect errors such as accessibility, exceeding axis limits, and singularity, and to avoid collisions and to fix errors. In order to do this, robotic fabrication simulation is created in the mixed-reality environment. The parameters required for the toolpath to be post-processed into robot code are determined by the user in the mixed-reality environment. Changes made to parameters can be monitored instantly in the holographic simulation created in the mixed-reality environment.

The robot code is sent to the industrial robot using the communication between the parametric-design program and the industrial robot. After receiving the robot code, the industrial robot executes the commands. If the user makes changes to the model parameters or robot code post-process parameters within the mixed-reality environment at the time of production, the following steps are automatically updated and the production process continues without interruption.

In order to create the proposed method, instant communication between the parametric-design program, the mixed-reality device, and the industrial robot control unit is required. Parameters of the model, geometry information of the model, robot code post-process parameters, and robot code data can be transmitted through instant communication. Figure 4 shows the communication diagram between the parametric-design software, the mixed-reality device, and the industrial robot.





**Figure 4.** Communication diagram between the parametric-modeling software, the mixed-reality device, and the industrial robot.

### 3.1. Communication and Simulation

In our study, five distinct software-development tasks were completed in order to create instant communication between the parametric modeling software, the mixed-reality device, and the industrial robot control unit, and to simulate the industrial robot in the mixed-reality device.

1. Running Grasshopper3d in “Headless Mode” and developing the REST API Server software for Grasshopper3d parametric modeling software;
2. Developing the REST API Client software in Unity Game Engine for HoloLens 2 Mixed-Reality Device;
3. Developing the inverse kinematic solver for 6-axis industrial robots with a spherical wrist in Unity Game Engine;
4. Developing the TCP Socket Server software for Kuka Robot Control Unit (KRC);
5. Developing TCP Socket Client Software in Unity Game Engine for HoloLens 2 Mixed-Reality device and Grasshopper3d parametric modeling software.

#### 3.1.1. REST API Server for Grasshopper3d Parametric Modeling Software

By default, the Grasshopper3d parametric-modeling tool is not accessible from other devices, such as a mobile device or a mixed-reality headset. The Grasshopper3d parametric-modeling tool runs only on the computer on which the program is installed. In our study, we developed an application programming interface (API), which enables users to access the Grasshopper 3d parametric-modeling tool via HTTP interface. Users can send input parameters with HTTP requests from the mixed-reality headset. The input parameters are calculated inside the Grasshopper3d parametric-modeling tool, and the results are returned with HTTP response to the program installed on the mixed-reality device, in near real-time.

REST API Server software has been developed for the Grasshopper3d program to instantly communicate with the mixed-reality device. Under REST architecture, the client and server can only interact in one way: the client sends a request to the server, and then the server sends a response back to the client. Servers cannot make requests and clients cannot respond. All interactions are initiated by the client. Incoming requests and outgoing responses are JSON formatted. JSON data packages are easy to parse and easy to generate with programming languages. C# programming language, .NET Framework, and NancyFX lightweight web framework [27] are preferred to develop the REST API Server.

In order for Grasshopper3d to respond to incoming requests, the Rhino.Inside feature that comes with the 7th version of the Rhino3d program has been extended. The Rhino.Inside is an open-source project that enables Rhino3d and Grasshopper3d programs to be used inside other programs running on the same computer such as Autodesk Revit, Autodesk AutoCAD, and Unity. The Rhino.Inside technology allows Rhino and Grasshopper to be embedded within other products. It may be possible starting Rhino

and Grasshopper as an add-in another product, to call directly into the host's native APIs from a Grasshopper or Rhino plug-in, to access Rhino's APIs through the host application; grasshopper definitions can be opened and previewed in Rhino within the same process as the parent, and objects can be natively created by Rhino or Grasshopper within the parent product [28].

In this study, primitive data types such as boolean, integer, double, string, and RhinoCommon SDK [29] data types including arc, box, circle, curve, line, mesh, mesh face, plane, point, rectangle, and vector were implemented and can be used as both input and output parameters for REST API Server communication requests and responses.

REST API Server software can be accessed through different client devices including a web browser, a mobile device, or other software. Figure 5 shows a sample Grasshopper3d definition and the generated result with parameters and Figure 6 shows HTTP request input parameters and the calculated result as HTTP response output parameters. In Figure 6, while receiving the HTTP request and returning the HTTP response, Grasshopper3d program runs in headless mode in the background.

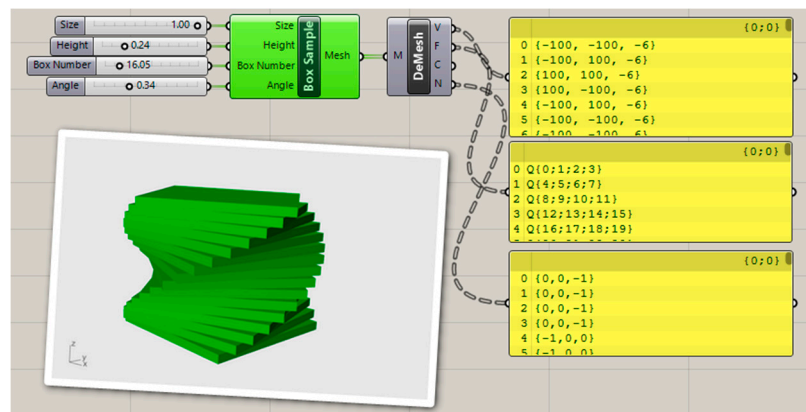


Figure 5. Sample Grasshopper3d definition.

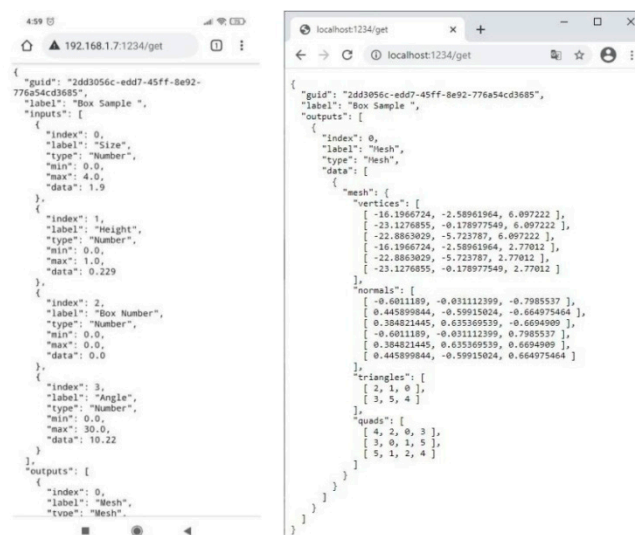


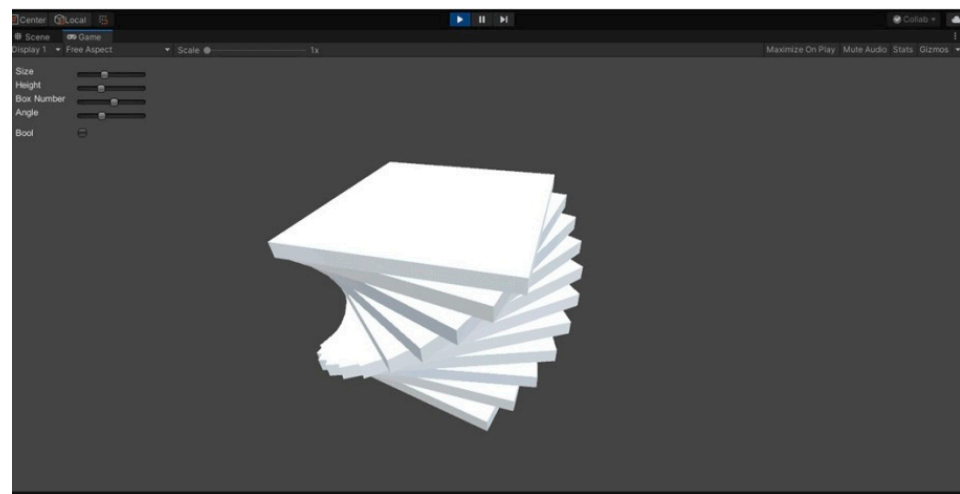
Figure 6. REST API Server sample request accessed through a mobile device (left) and sample response accessed through a web browser (right) running Grasshopper3d definition in headless mode.

### 3.1.2. REST API Client for HoloLens 2 Mixed-Reality Device

In the next step of the study, the REST API client software that sends requests to the REST API Server and receives the responses was developed for the mixed-reality device.

The Unity Game Engine and Mixed-Reality Toolkit (MRTK) [30] were used to develop the REST API client software for the mixed-reality device.

The Unity Game Engine has the right-handed Y-Up coordinate system whereas Grasshopper3d has the left-handed Z-Up coordinate system. Grasshopper primitive and RhinoCommon SDK [29] data types retrieved from REST API Server program are converted to Unity data types and Unity coordinate system. In this study, arc, boolean, box, circle, curve, integer, line, mesh, float, plane, point, rectangle, string, and vector data types were supported in Unity Game Engine and Mixed-Reality Toolkit. Figure 7 shows the REST API Client program running inside Unity Game Engine. If the user changes *size*, *height*, *box number*, or *rotation angle* parameters, the Unity Game Engine sends these parameters to Grasshopper3d modeling tool via HTTP request and gets the calculated result as an HTTP response. In Figure 7, the boxes are generated inside Grasshopper3d parametric-design tool with the parameters sent over HTTP communication.



**Figure 7.** Box model mesh data are generated inside Grasshopper3d parametric-modeling tool using *size*, *height*, *box number*, and *rotation angle* parameters (upper left corner).

### 3.1.3. Inverse Kinematic Solver for 6-Axis Industrial Robots

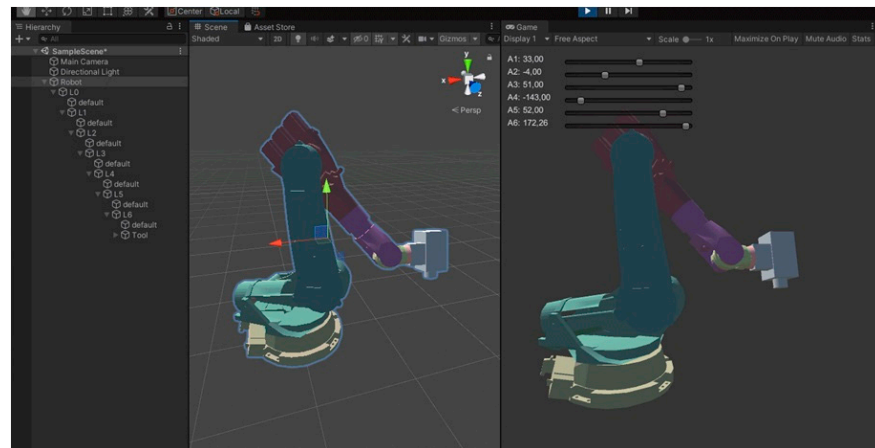
In this study, an inverse kinematic solver of 6R serial industrial robot manipulators with an Euler wrist was developed for the Unity Game Engine, which has the right-handed Y-Up coordinate system. For an industrial robot, inverse kinematics refers to solving angular values of its joints to reach a given desired position and orientation value. In this way, a 6-six-axis industrial robot with a spherical wrist can be simulated in mixed-reality environment. Simulating the industrial robot is important for detecting singularities, reachability errors, exceeding angular limits, and collision detection. Figure 8 shows Kuka KR210 simulation inside the Unity Game Engine.

### 3.1.4. TCP Socket Server Software for Kuka Robot Control Unit (KRC)

In the next step, TCP Socket Server software was developed for the industrial robot. Unlike REST API communication, TCP Socket communication is a two-way communication. Using TCP Socket communication, the industrial robot receives robot commands, executes, and sends the result back. Execution time is needed between receiving the robot commands and sending the results back.

Kuka KR210 industrial robot was used in this study. Since the Windows 95 operating system was installed on the VKRC2 robot control unit of the KR210 industrial robot, Visual Basic 6.0 programming language was used while developing the TCP Socket Server software. Figure 9 shows TCP Socket Server software screenshot taken from Kuka Control Robot Unit (VKRC2).





**Figure 8.** Kuka KR210 simulation inside Unity Game Engine.



**Figure 9.** TCP Socket Server software screenshot on Kuka Control Robot Unit (VKRC2).

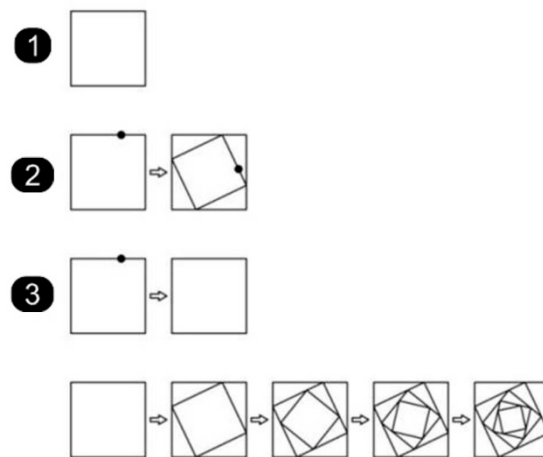
### 3.1.5. TCP Socket Client Software for HoloLens 2 Mixed-Reality Device and Grasshopper3d Parametric Modeling Software

In this study, TCP Socket client software was developed for the HoloLens 2 Mixed-Reality device and the Grasshopper3d parametric-modeling tool application programming interface. In this way, the industrial robot receives robot commands, executes, and sends reports to the mixed-reality device and the Grasshopper3d parametric modeling software runs in headless mode.

### 3.2. Shape Grammars

Shape Grammars were first invented by George Stiny and James Gips in their 1972 article *Shape Grammars and the Generative Specification of Painting and Sculpture* [31]. Shape grammars are rule systems of transformational shape rules that describe the design of a shape. A shape rule defines how an existing (part of a) shape can be transformed [32].

Shape grammars consist of an initial shape which can be a point, line, or polygon; a start rule; transformation rules, which are usually applied recursively; and a termination rule. Figure 10 shows the initial shape, shape rules for a standard shape grammar, and the results generated by applying the transformation rules recursively [32].



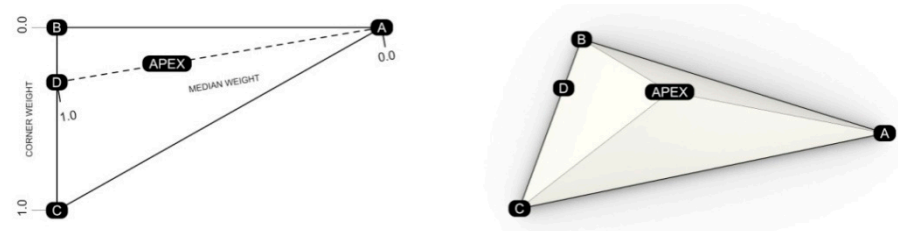
**Figure 10.** Standard shape grammar—initial shape (1), transformation rule (2), termination rule (3), and results generated by applying the transformation rules recursively [32].

#### 4. Results

In order to test the proposed method in this study, a robotic-fabrication-workshop test environment is created. The proposed method was tested on a design product, which was defined with the shape-grammar method using parametric-modeling tools. Natural stone material was chosen to test the proposed method in robotic fabrication.

In the study, the standard shape-grammar method was used to generate the three-dimensional design product in parametric-design software. Triangular areas were converted into triangular pyramids. The locations of the *apex points* of these triangular pyramids were calculated with *median-weight*, *corner-weight*, and *height* parameters.

In a triangle defined by the A, B, and C corner points, the location of the D point was calculated with the *corner-weight* parameter between the B and C points. Then, the location of the *apex point* was calculated with the *median-weight* parameter between A and D points and the *height* parameter. Figure 11 shows the *apex point* and the *corner-weight* and *median-weight* parameters. Figure 12 shows the results generated by applying the transformation rules and the termination rule.

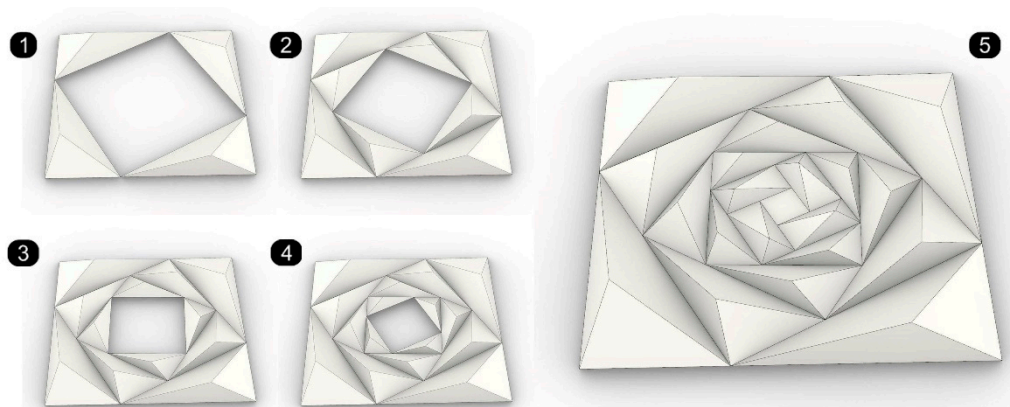


**Figure 11.** Apex point and *corner-weight* and *median-weight* parameters.

Figure 13 shows the results generated by applying different transformation rules defined with *corner-weight*, *median-weight*, *height*, *rotation*, and *repeat* parameters, and different termination rules.

In the study, a natural stone robotic fabrication workshop was created to test the proposed method. Figure 14 shows that the user can change the parameters of parametric-design and robotic-fabrication tasks within the mixed-reality environment and robotic fabrication can continue uninterrupted.

Figure 15 shows that the user can access and change the parametric design and robotic-fabrication parameters using the mixed-reality device. Figure 16 shows that the user can change design and production parameters, and gets instant visual and spatial feedback of the design and production alternatives while robotic fabrication continues. The following tasks in the workflow do not need to be repeated in the production phase.



**Figure 12.** The results, generated by applying the transformation rule recursively (1–4) and the termination rule (5).



**Figure 13.** The results, generated by applying different transformation rules, defined with *corner-weight*, *median-weight*, *height*, *rotation*, and *repeat* parameters, and different termination rules.

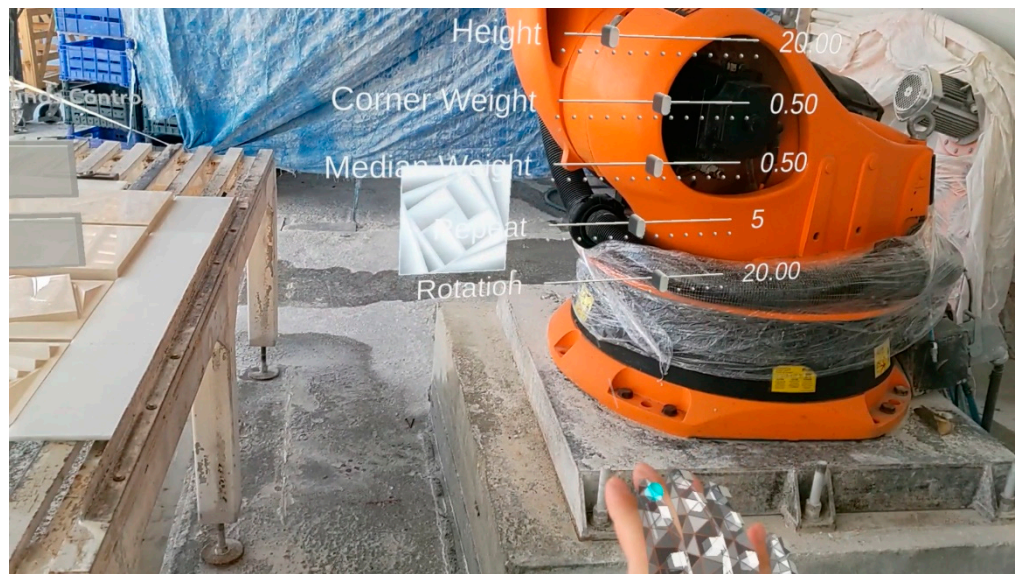


**Figure 14.** The user can change the parameters of parametric-design and robotic-fabrication tasks within mixed-reality environment.





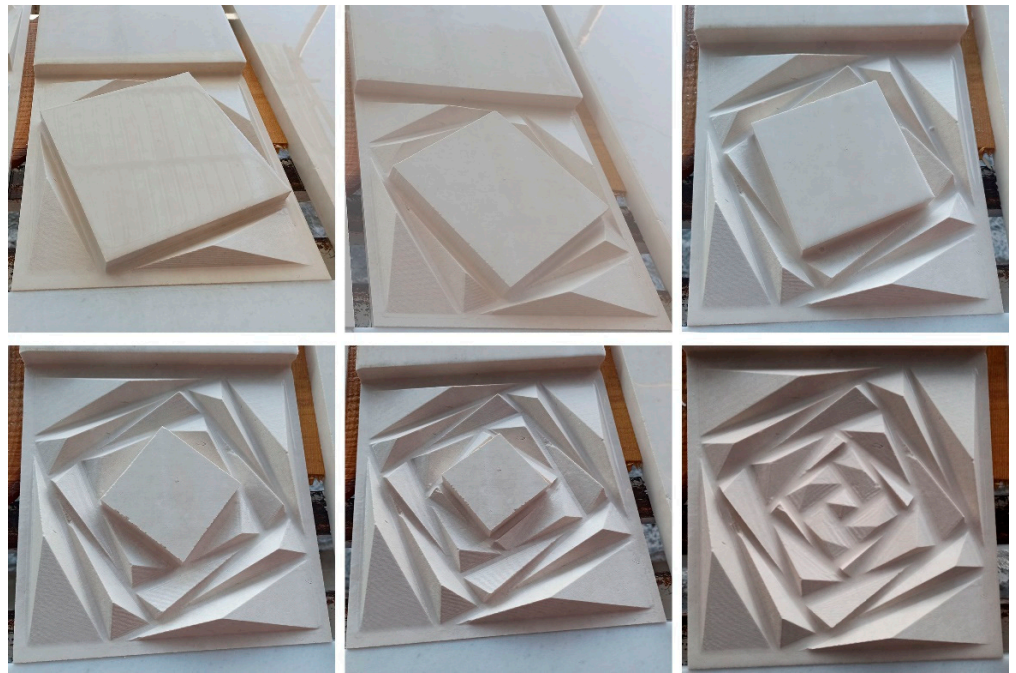
**Figure 15.** The user can access and change the parametric-design and robotic-fabrication parameters using the mixed-reality device.



**Figure 16.** Interactive parametric design and interactive robotic fabrication controlled with the mixed-reality device in the production phase.

The user changes the parameters of the shape-grammar transformation rule at each iteration. Figure 17 shows the design product that was manufactured with the proposed method. The results of each iteration, generated by applying different transformation rules, defined with *corner-weight*, *median-weight*, *height*, and *rotation* parameters, and the result of the termination rule at the last iteration can be seen in Figure 17.

Figure 18 shows the production results of design products defined with the shape-grammar method. There are nine different natural stone products in the figure. Eight production results were manufactured with existing methods using parametric-modeling tools. The product located at the center was manufactured with the proposed method, and different transformation rules (*corner weight*, *median weight*, *height*, and *rotation*) were applied to this product at each iteration, while production continued. Thus, transformation rules were irregular, unlike the other eight pieces in the figure.



**Figure 17.** The results of each iteration, generated by applying different transformation rules, defined with *corner-weight*, *median-weight*, *height*, and *rotation* parameters at each iteration, and the result of the termination rule at the last iteration.



**Figure 18.** The result, generated with proposed method by applying different transformation rules defined with *corner-weight*, *median-weight*, *height*, and *rotation* parameters at each iteration (**center**) and the results generated by applying same transformation rules at each iteration (**others**).



The design and robotic-fabrication processes of the proposed method are shown. The proposed method and the existing methods are compared and discussed in terms of mass-customization, the design-to-production process, scalability, machine time, process, and material efficiency, human–robot collaboration, production-immanent modeling, interactive design, and interactive robotic-fabrication possibilities. In Table 1, the robotic fabrication offline programming method, programming with parametric robot control tools, and the proposed method are compared in terms of design and robotic-fabrication possibilities based on the observations obtained from the test results.

**Table 1.** Comparison chart of robotic-fabrication offline programming, parametric robot-control tools, and the proposed method.

	Offline Programming	Parametric Robot Control Tools	Proposed Method
Users need to work with CAD/CAM tools	Yes	No	No
Industrial robot offline programming knowledge is required	Yes	No	No
Usage of the method is limited by parametric-modeling tools	No	Yes	Yes
Production-immanent modeling tools are offered	No	Yes	Yes
Mass-customization tools are offered	No	Yes	Yes
Parametric design and robotic fabrication are discrete tasks in the workflow	Yes	Yes	No
Users can explore and change design and production parameters in the design phase	No	Yes	Yes
Users can explore and change design and production parameters in the production phase	No	No	Yes
Users can get visual and spatial feedback on the design and production alternatives in the design phase	No	Yes	Yes
Users can get visual and spatial feedback on the design and production alternatives in the production phase	No	No	Yes
If users change the design or fabrication parameters, the following tasks in the workflow must be repeated, before production	Yes	No	No
If users change the design or fabrication parameters, the following tasks in the workflow must be repeated, while production continues	Yes	Yes	No
Users can interact with the design and fabrication parameters, while robotic fabrication continues	No	No	Yes
Material efficiency by monitoring the changes on both stock materials and design products, before production	No	Yes	Yes
Material efficiency by monitoring the changes on both stock materials and design products, while production continues	No	No	Yes
Multiple users can collaborate in the design and robotic fabrication by dynamically exploring design and production alternatives, before production	No	Yes	Yes
Multiple users can collaborate in the design and robotic fabrication by dynamically exploring design and production alternatives, while production continues	No	No	Yes

## 5. Discussion and Future Work

The proposed method and other existing methods are compared and discussed in terms of design and robotic-fabrication possibilities based on the observations obtained from the test results. With the proposed method, the user can explore design and production

alternatives within the mixed-reality environment by changing the parameters, and gets instant visual and spatial feedback on the design and production alternatives. With changing the parameters, the design product as well as the robot code required for the production of these parts are updated, and the robot code is uploaded to the industrial robot instantly. These tasks are completed in one unified step and the design-to-production process is shortened since the user does not need to do manual interventions in the intermediate steps. Robotic fabrication can continue uninterrupted with human–robot collaboration.

Different from existing robotic fabrication workflows, with the proposed method, users can change the design and fabrication parameters while robotic fabrication continues. The design and manufacturing processes are combined and blended, thus users can complete the design and manufacturing tasks within one unified framework. Unlike other existing robotic fabrication methods, the proposed method provides interactive robotic-fabrication possibilities in addition to interactive parametric-design possibilities.

In existing robotic-fabrication workflows, parametric design and robotic fabrication are discrete operations. If users want to make changes in the design phase, or production phase the robot code generated on the computer needs to be transferred and uploaded again to the robot control unit because the outputs of the previous steps are used as inputs for the next steps. Users may need to work with different software CAD/CAM tools and repeat these steps on both the computer and the robot control unit. Unlike other methods, in the proposed method, parametric-design and robotic-fabrication possibilities are offered to the user as one unified step within the mixed-reality environment and the time required to complete the design and manufacturing process is shortened. In addition, with this improved workflow, industrial robot-programming knowledge is not required to complete robotic-fabrication tasks.

Another advantage of the proposed method is that users can use stock material resources more effectively. The digital twin of both parametric design and robotic fabrication is created and users can monitor the changes in both stock materials and design products in the mixed-reality environment while robotic fabrication continues. Thus, users can use stock material resources more effectively with the proposed method.

The proposed method allows using parametric-modeling tools within the mixed-reality environment in both the design and production phases of robotic fabrication. This allows users to perform robotic fabrication interactively. In this interactive robotic fabrication, users can both use the design and production possibilities offered by parametric-modeling tools such as mass-customization in the production phase, as well as access design opportunities such as interactive design, emergent design, and generative design in the design phase. However, the usage of the proposed method is limited with parametric-modeling tools.

In addition, the proposed method allows multiple users to co-exist in the same mixed-reality environment and interact with real and virtual objects at the same time. Thus, parametric design and robotic fabrication can be performed by multiple users and with multiple industrial robots. Design and production alternatives can be explored by multiple users. In this respect, the method can be scaled in terms of the number of users, the number of industrial robots used in production, and human–robot collaboration.

There are future studies to be done on exploring the potential of the proposed method improved with computer vision and machine-learning technologies. For future studies, the research team focused on improving the proposed method with image-tracking and object-tracking technologies provided by augmented reality development toolkits [33,34].

**Author Contributions:** Y.B., conceptualization, methodology, software, validation, resources, writing—original draft preparation, project administration, and funding acquisition; G.Ç., conceptualization, methodology, writing—review and editing, project administration, supervision, and funding acquisition. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Istanbul Technical University, Scientific Research Projects Coordination Unit. Project Number: MDK-2020-42387.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Steinhagen, G.; Braumann, J.; Brüninghaus, J.; Neuhaus, M.; Brell-Cokcan, S.; Kuhlenkötter, B. Path planning for robotic artistic stone surface production. In *Robotic Fabrication in Architecture, Art and Design 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 122–135.
2. Brugnaro, G.; Hanna, S. Adaptive robotic training methods for subtractive manufacturing. In Proceedings of the 37th annual conference of the association for computer aided design in architecture (ACADIA), Cambridge, MA, USA, 2–4 November 2017; pp. 164–169.
3. Parascho, S.; Gandia, A.; Mirjan, A.; Gramazio, F.; Kohler, M. *Cooperative Fabrication of Spatial Metal Structures*; ETH Library: Zürich, Switzerland, 2017; pp. 24–29.
4. Jahn, G.; Wit, A.J.; Pazzi, J. [BENT] Holographic handcraft in large-scale steam-bent timber structures. *ACADIA 2019*.
5. Gozen, E. A Framework for a Five-Axis Stylus for Design Fabrication. Architecture in the Age of the 4th Industrial Revolution. In Proceedings of the 37th eCAADe and 23rd SIGRaDi Conference-Volume 1, University of Porto, Porto, Portugal, 11–13 September 2019; pp. 215–220. [\[CrossRef\]](#)
6. Goepel, G.; Crolla, K. Augmented Reality-based Collaboration-ARgan, a bamboo art installation case study. In Proceedings of the 25th International Conference of the Association for Computer-Aided Architectural Design Research in Asia, Bangkok, Thailand, 5–6 August 2020.
7. Jahn, G.; Newnham, C.; van den Berg, N.; Iraheta, M.; Wells, J. Holographic Construction. In *Design Modelling Symposium Berlin*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 314–324.
8. Fazel, A.; Izadi, A. An interactive augmented reality tool for constructing free-form modular surfaces. *Autom. Constr.* **2018**, *85*, 135–145. [\[CrossRef\]](#)
9. Jahn, G.; Newnham, C.; van den Berg, N.; Beanland, M. Making in mixed reality. In Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA), Mexico City, Mexico, 18–20 October 2018; pp. 88–97, ISBN 978-0-692-17729-7. [\[CrossRef\]](#)
10. Sun, C.; Zheng, Z. Rocky Vault Pavilion: A Free-Form Building Process with High Onsite Flexibility and Acceptable Accumulative Error. In Proceedings of the International Conference on Computational Design and Robotic Fabrication, Shanghai, China, 7–8 July 2019; Springer: Singapore, 2019; pp. 27–36.
11. Wibranek, B.; Tessmann, O. Digital Rubble Compression-Only Structures with Irregular Rock and 3D Printed Connectors. In Proceedings of the IASS Annual Symposia. International Association for Shell and Spatial Structures (IASS), Barcelona, Spain, 7–10 October 2019; Volume 2019, pp. 1–8.
12. Yue, Y.T.; Zhang, X.; Yang, Y.; Ren, G.; Choi, Y.K.; Wang, W. Wiredraw: 3d wire sculpturing guided with mixed reality. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 3693–3704.
13. Hahm, S.; Maciel, A.; Sumitomo, E.; Lopez Rodriguez, A. FlowMorph-Exploring the human-material interaction in digitally augmented craftsmanship. In Proceedings of the 24th CAADRIA Conference-Volume 1, Victoria University of Wellington, Wellington, New Zealand, 15–18 April 2019; pp. 553–562. [\[CrossRef\]](#)
14. Betti, G.; Aziz, S.; Ron, G. Pop Up Factory: Collaborative Design in Mixed Reality-Interactive live installation for the makeCity festival, 2018 Berlin. In Proceedings of the eCAADe + SIGRaDi 2019, Porto, Portugal, 11–13 September 2019.
15. Morse, C.; Martinez-Parachini, E.; Richardson, P.; Wynter, C.; Cerone, J. Interactive design to fabrication, immersive visualization and automation in construction. *Constr. Robot.* **2020**, *4*, 163–173. [\[CrossRef\]](#)
16. Peng, H.; Briggs, J.; Wang, C.Y.; Guo, K.; Kider, J.; Mueller, S.; Baudisch, P.; Guimbretière, F. RoMA: Interactive fabrication with augmented reality and a robotic 3D printer. In Proceedings of the 2018 CHI conference on human factors in computing systems, Montreal, QC, Canada, 21–26 April 2018; pp. 1–12.
17. Chang, T.W.; Hsiao, C.F.; Chen, C.Y.; Huang, H.Y. CoFabs: An Interactive Fabrication Process Framework. In *Architectural Intelligence*; Springer: Singapore, 2020; pp. 271–292.
18. Johns, R.L.; Anderson, J.; Kilian, A. Robo-Stim: Modes of human robot collaboration for design exploration. In *Design Modelling Symposium Berlin*; Springer: Cham, Switzerland, 2019; pp. 671–684.
19. Kyjaneck, O.; Al Bahar, B.; Vasey, L.; Wannemacher, B.; Menges, A. Implementation of an augmented reality AR workflow for human robot collaboration in timber prefabrication. In Proceedings of the 36th International Symposium on Automation and Robotics in Construction, Banff, AB, Canada, 21–24 May 2019.
20. Amtsberg, F.; Yang, X.; Skoury, L.; Wagner, H.J.; Menges, A. iHRC: An AR-based interface for intuitive, interactive and coordinated task sharing between humans and robots in building construction. In Proceedings of the International Symposium on Automation and Robotics in Construction, Dubai, United Arab Emirates, 2–4 November 2021; IAARC Publications: Corvallis, OR, USA, 2021; Volume 38, pp. 25–32.
21. Eswaran, M.; Bahubalendruni, M.R. Challenges and opportunities on AR/VR technologies for manufacturing systems in the context of industry 4.0: A state of the art review. *J. Manuf. Syst.* **2022**, *65*, 260–278. [\[CrossRef\]](#)
22. Inkulu, A.K.; Bahubalendruni, M.R.; Dara, A.; SankaranarayanaSamy, K. Challenges and opportunities in human robot collaboration context of Industry 4.0-a state of the art Review. *Ind. Robot. Int. J. Robot. Res. Appl.* **2021**. [\[CrossRef\]](#)

23. Brell-Cokcan, S.; Braumann, J. A New Parametric Design Tool for Robot Milling. In Proceedings of the 30th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA), New York, NY, USA, 21–24 October 2010; pp. 357–363.
24. Braumann, J.; Brell-Cokcan, S. Parametric Robot Control: Integrated CAD/CAM for Architectural Design. In Proceedings of the 31st Annual Conference of the Association for Computer Aided Design in Architecture, Calgary, AB, Canada, 18–20 September 2013; pp. 242–251.
25. Schwartz, T. HAL: Extension of a visual programming language to support teaching and research on robotics applied to construction. In *Robotic Fabrication in Architecture, Art and Design*; Brell-Cokcan, S., Braumann, J., Eds.; Springer: Vienna, Austria, 2012; pp. 92–101.
26. Microsoft Hololens 2 Mixed-Reality Device. Available online: <https://www.microsoft.com/en-us/hololens> (accessed on 1 July 2022).
27. Nancy Is a Lightweight Framework for Building HTTP Based Services on NET and Mono. Available online: <https://nancyfx.org/> (accessed on 1 July 2022).
28. Rhino. Inside Technology. Available online: <https://github.com/mcneel/rhino.inside> (accessed on 1 July 2022).
29. RhinoCommon. Available online: <https://developer.rhino3d.com/guides/rhinocommon/what-is-rhinocommon/> (accessed on 1 July 2022).
30. Mixed-Reality Toolkit Documentation. Available online: <https://docs.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/> (accessed on 1 July 2022).
31. Stiny, G.; Gips, J. Shape grammars and the generative specification of painting and sculpture. In Proceedings of the IFIP Congress, Ljubljana, Yugoslavia, 23–28 August 1971; Volume 2, pp. 125–135.
32. Stiny, G. Introduction to shape and shape grammars. *Environ. Plan. B Plan. Des.* **1980**, *7*, 343–351. [[CrossRef](#)]
33. ARCore—Google Developers. Available online: <https://developers.google.com/ar> (accessed on 1 July 2022).
34. ARKit—Apple Developer. Available online: <https://developer.apple.com/augmented-reality/arkit/> (accessed on 1 July 2022).