From CAD Models to Toy Brick Sculptures: A 3D Block Printer*

Yusuke Maeda[1], Ojiro Nakano[2], Takashi Maekawa[1] and Shoji Maruo[1]

***Abstract*— This paper presents a robotic 3D printer: a robot system that can assemble toy brick sculptures from their 3D CAD models. In this system, a 3D CAD model is automatically converted to a block model consisting of primitive toy blocks. Then an assembly plan of the block model is automatically generated, if feasible. According to the plan, an industrial robot assembles a brick sculpture layer by layer from bottom to top. We demonstrate successful assembly of several brick sculptures.**

I. Introduction

Recent progress of additive manufacturing technology attracts considerable attention not only in the engineering community but also in the public. Additive manufacturing processes can be classified into several groups: fused deposition modeling, stereolithography, selective laser sintering, and so on [1].

Building blocks like LEGO can be used to form arbitrary shapes [2]. This is a kind of additive manufacturing, too. Digital materials including LEGO-like blocks have favorable properties for manufacturing [3]. In this paper, we present a robot system that assembles LEGO-like toy blocks to produce brick sculptures.

This system can convert a 3D CAD model into its corresponding block model consisting of primitive toy blocks that are smaller than LEGO ("nanoblock" by Kawada) automatically. Then the block model is analyzed to generate its assembly plan. If a feasible plan is obtained, an industrial robot assembles blocks to produce a toy brick sculpture (Fig. 1). We show the details of the system and some successful assemblies of block sculptures.

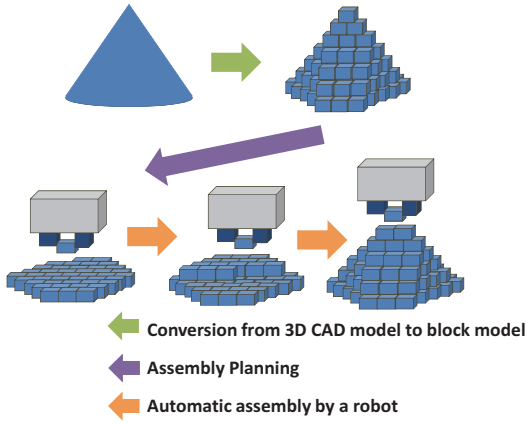


Fig. 1: Overview of 3D Block Printing

*This work was supported by JSPS KAKENHI 15H03944.

[1]Yusuke Maeda, Takashi Maekawa and Shoji Maruo are with Faculty of Engineering, Yokohama National University, 79-5 Tokiwadai, Hodogaya-ku, Yokohama, JAPAN maeda@ynu.ac.jp

[2]Ojiro Nakano is with Graduate School of Engineering, Yokohama National University

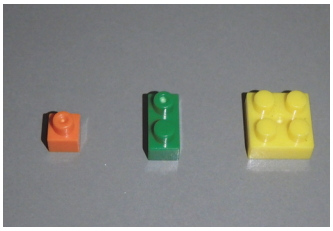


Fig. 2: 1×1, 1×2 and 2×2 nanoblocks

II. Related Works

Robots that assemble LEGO blocks are popular demonstrations in exhibitions. A LEGO-made machine that can assemble LEGO sculptures was also developed [4]. However, in most cases, the robots are programmed in advance to assemble specific toy brick sculptures.

Hiller and Lipson analyzed "voxel printing" in detail and mentioned its advantages including perfect repeatability, multiple materials and smart voxels [3]. They also developed a "VoxJet" printer using spherical voxels [5].

Medellin et al. presented a system that can assemble cubes to form shapes [6]. A 3D CAD model is converted to an octree representation and a SCARA robot assembles cubes that correspond to the octree with gluing.

Sekijima et al. proposed to use truncated octahedra with magnetic connectors ("Kelvin blocks") to form shapes [7]. They developed an assembly robot with parallel mechanism for Kelvin blocks.

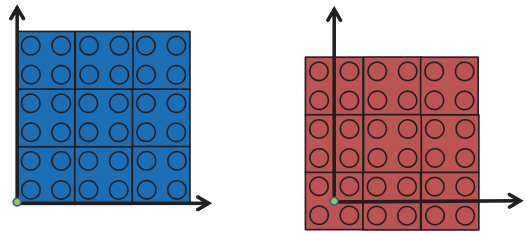
III. Building Blocks

We use "nanoblock" by Kawada as building blocks (Fig. 2). The toy brick has a LEGO-like shape but smaller than LEGO blocks. Its smallest unit has edges of 4 [mm] (W) $\times$ 4 [mm] (D) $\times$ 3 [mm] (H). We denote the horizontal unit length (4 [mm]) by 1 [unit]. In this study, 1×1, 1×2, and 2×2 nanoblocks with the minimum height are used as primitive building blocks; we specify block sizes by the number of rows $\times$ columns of studs. 1×2 blocks can be transposed to 2×1 in assembly.

We assemble the building blocks layer by layer from bottom to top. The layers are numbered consecutively from 1 (the bottom) to n (the top).

IV. From CAD Models to Block Models

There are many previous studies on conversion from 3D models to LEGO models. A survey on them is found in [8].



(a) Odd Layer (b) Even Layer

Fig. 3: Shifted Replacement

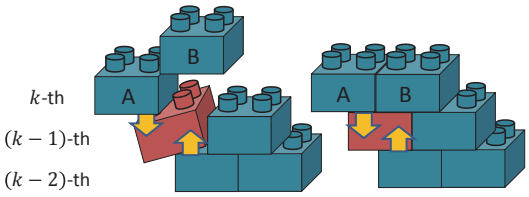


Fig. 4: Assembling Order: A→B and B→A

In this study, we used an algorithm developed by Kozaki et al. [9] for conversion from CAD models to block models. A 3D CAD model in OFF (Object File Format) is converted to unit voxels with edge length ratio of 4:4:3, which is different from those for LEGO blocks (5:5:6 for their smallest unit). Then the voxels are replaced by the primitive blocks so that the number of blocks are reduced using simulated annealing. The reference point for the replacement is shifted one unit in column and row directions for each layer to strengthen the assembled structure (Fig. 3).

V. Overview of Assembly Planning

A. Assembly Failures

In order to show the importance of assembly planning, here we describe an example of assembly failure. Let us consider placing blocks in the k-th layer. As shown in the left case of Fig. 4, when we try to place the blocks in the order of A and B, the orange block in the $(k-1)$-th layer rotates in the process of placement of Block A and assembly fails. Note that this rotation is caused even under perfect position control of Block A because the force required to insert Block A into studs is larger than the maximum force that the orange block can resist. On the other hand, when we try to place the blocks in the order of B and A (the right case of Fig. 4), the placement of Block B strengthens the scaffold structure including the orange block. Therefore the insertion of Block A afterward does not cause the rotation of the orange block and assembly succeeds.

This example illustrates that the order of block placement is important. We describe the details of assembly planning below.

B. Assembly Strategy

As we described above, our system places blocks layer by layer from bottom to top. The assembly planning in this study can be divided into the following subproblems:

1) *Block Assemblability Check*: We have to judge whether a block placement is safe or may collapse the block structure.
2) *Block Placement Ordering*: We have to decide an appropriate order of safe block placements in each layer.

Our assembly strategy in the k-th layer ($k > 2$) can be outlined as follows:

1) Check the assemblability of all the blocks to be placed in the k-th layer. If there is a floating block, which has no connections to the $(k-1)$-th layer, give up.
2) If no blocks are assemblable in the k-th layer, give up. Otherwise, place an assemblable block that has the highest priority.
3) If all the blocks in the k-th layer have been assembled, proceed to the next layer. Otherwise, update the result of block assemblability check according to the previous placement, and go back to 2).

Note that the assemblability check is unnecessary for the first (bottom) layer and the second layer except for floating blocks. The details of the procedure is described in the following sections.

VI. Block Assemblability Check

Let us describe how to check the assemblability of a block in the k-th layer. The block must have one or more connections to studs of blocks in the $(k-1)$-th layer. We focus on the placement of a block in the k-th layer on a single block in the $(k-1)$-th layer. If a block in the k-th layer is placed on multiple blocks in the $(k-1)$-th layer, we check the assemblability for each combination.

A. Contact Region Inclusion

Let us denote the contact region between the k-th and $(k-1)$-th layers by $U_{k,k-1}$. We also define a projection operator $P()$ that projects a region to horizontal X-Y plane. If the following holds as in the case of Fig. 5a, the placement of the green block in the k-th layer does not cause the rotation of the orange block in the $(k-1)$-th region:

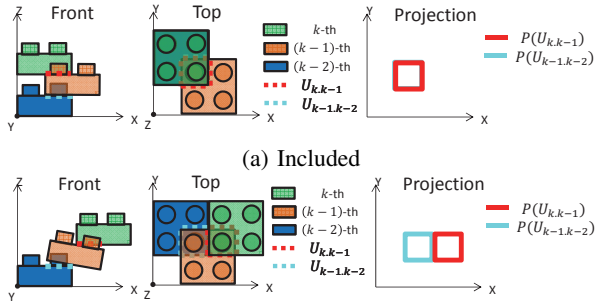
$$P(U_{k,k-1}) \subseteq P(U_{k-1,k-2}). \quad (1)$$

This is because the vertical contact forces exerted at $U_{k,k-1}$ can be canceled by contact forces exerted at $U_{k-1,k-2}$. On the other hand, in the case of Fig. 5b, (1) does not hold and the orange block in the $(k-1)$-th layer may be rotated.

Thus if (1) holds, the block in the k-th layer can be placed on the block in the $(k-1)$-th layer. We call such placements "Case A."

B. Contact Edge Inclusion

Let us consider a case as shown in Fig. 6. In this case, (1) does not hold for the green block in the k-th layer. However, due to the shape of nanoblocks (Fig. 7), the contact forces between the block in the k-th and $(k-1)$-th layers are exerted only at a part of the bottom edges of the green block, $E_{k,k-1} (\subseteq U_{k,k-1})$. Moreover, the support polygon of the orange block in the $(k-1)$-th layer, denoted by $S_{k-1,k-2}$,



(a) Included

(b) Not Included

Fig. 5: Contact Region Inclusion

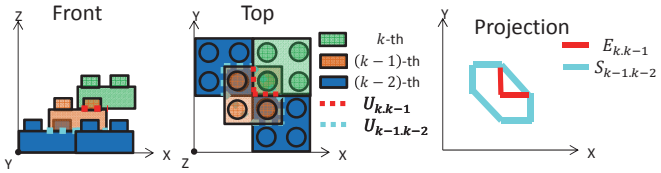


Fig. 6: Contact Edge Inclusion

includes $U_{k-1,k-2}$. Thus if the following holds, the green block can be placed on the orange block:

$$P(E_{k,k-1}) \subseteq P(S_{k-1,k-2}). \tag{2}$$

This is because the vertical contact forces exerted at $E_{k,k-1}$ can be canceled by contact forces exerted at $S_{k-1,k-2}$. We call such placements "Case B."

C. Multi-Stud Connection

Let us consider the case as shown in Fig. 8a. In this case, the green block in the k-th layer has two-stud connection with the orange block in the $(k-1)$-th layer. Although neither (1) nor (2) holds,

$$P(U_{k,k-1}) \cap P(U_{k-1,k-2}) \neq \emptyset. \tag{3}$$

Empirically we found that the blocks with such a multi-stud connection are assemblable when (3) holds except for the cases where all the following are satisfied:

- The block in the $(k-1)$-th layer under the placed block is 2×2.
- The placed block in the k-th layer is 2×2 with four-stud connection with the block in the $(k-1)$-th layer (Fig. 8b), or 1×2 with two-stud connection (Fig. 8c).
- $\mathrm{Area}(P(U_{k,k-1}) \cap P(U_{k-1,k-2})) = 1$, where $\mathrm{Area}()$ denotes the area of the region [unit^2].

We call such placements with multi-stud connections "Case E" with the above exceptions.

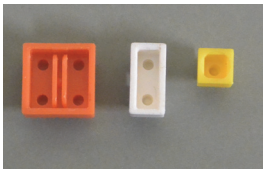
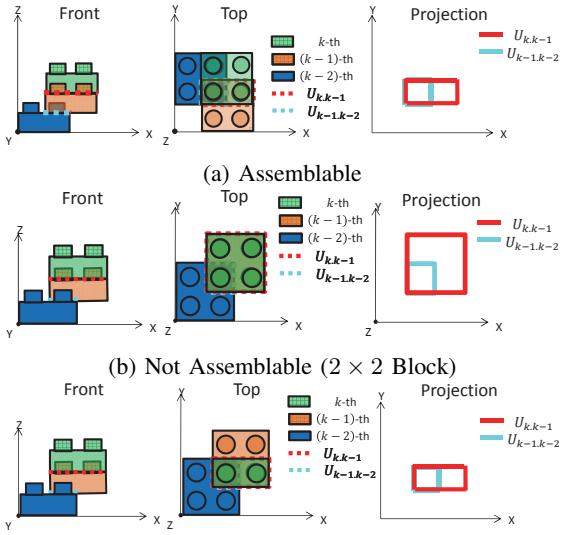


Fig. 7: Bottom Sides of Blocks



(a) Assemblable

(b) Not Assemblable (2×2 Block)

(c) Not Assemblable (1×2 Block)

Fig. 8: Multi-Stud Connection

D. Change of Block Assemblability

Let us consider the case as shown in Fig. 9a. In this case, neither (1) nor (2) holds and therefore the green block in the k-th layer is not assemblable. However, after we place another adjacent block in the k-th layer as shown in Fig. 9b, it becomes possible to place the block that was not assemblable.

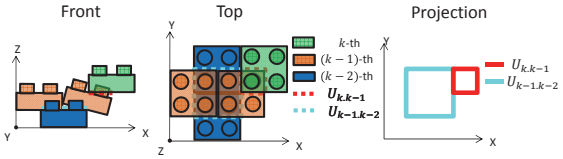
The change of the block assemblability is partially due to the stronger scaffold structure made by the new connection between the blocks in the $(k-1)$-th layer through the other block placement in the k-th layer. Moreover, the side contact between the blocks in the k-th layer also helps to avoid collapse of the structure; the rotation of the blocks in the $(k-1)$-th layer is accompanied by the rotation of the other block in the k-th layer, which is prevented by the side contact.

Thus we judge the situations like the case of Fig. 9b as assemblable, where there is an adjacent block in the k-th layer that makes connection between the blocks in the $(k-1)$-th layer under the placed block. We call this situation "Case D" for previously unassemblable blocks. If the block to be placed was previously in "Case E," we call this situation "Case C," which should be safer.

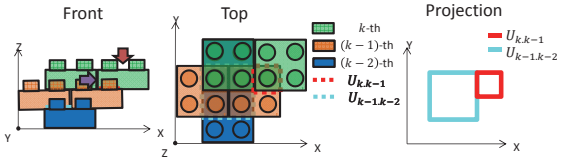
E. Example of Block Placement

As described above, feasible block placements are categorized into Case A through E. They are named in order of descending safety. Additionally we denote the other placements by "Case X." The success rate of Case-X placements is not high and therefore we regard Case X as infeasible.

Here we present an example as shown in Fig. 10a. In this case, the green block in the k-th layer should be placed on the three orange blocks in the $(k-1)$-th layer. The number 1 connection is categorized as Case X (Fig. 10b). On the other hand, the number 2 and 3 connections satisfy (1) (Fig. 10c and 10d). Therefore they are categorized as Case A. Thus

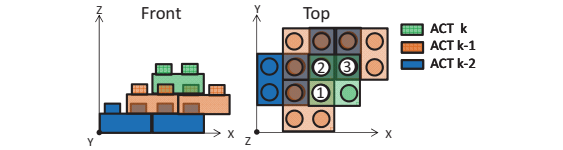


(a) Unassemblable Block

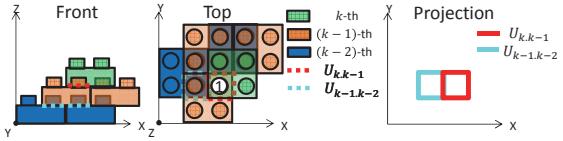


(b) Judgment Change by Another Placement

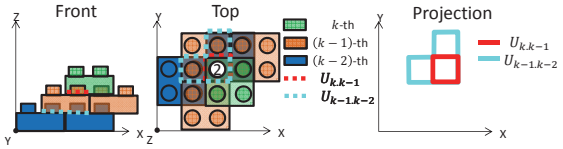
Fig. 9: Change of Block Assemblability



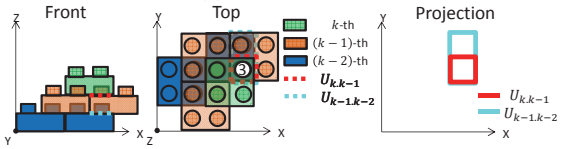
(a) Block Placement with Three Connections



(b) Connection 1



(c) Connection 2



(d) Connection 3

Fig. 10: Example of Block Placement

the placement of the green block in the k-th layer has two Case-A and one Case-X connections.

VII. Block Placement Ordering

In order to assemble blocks in the k-th layer, the order of the block placements must be determined. We use a priority-based ordering for this purpose.

A. Indices for Block Prioritization

Here we define the following indices for each block in the k-th layer ("Block A") for block prioritization:

- n_b: the number of the blocks in the $(k-1)$-th layer with which Block A connects.
- n_c: the number of the studs of the blocks in the $(k-1)$-th layer with which Block A connects.

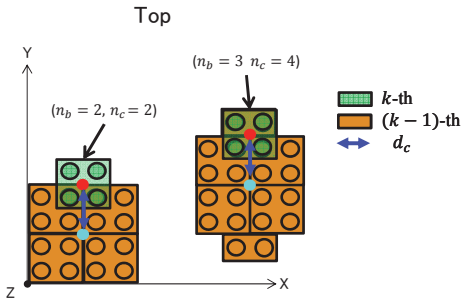


Fig. 11: Distances between Centroids

- d_c: the horizontal distance between the centroid of Block A and the centroid of the connected figure made of the blocks in the $(k-1)$-th layer on which Block A is placed.

Fig. 11 illustrates examples of the indices. For the green block in the k-th layer in the left of Fig. 11, it is placed on two 2×2 blocks with one-stud connection for each. Thus $n_b = 2$, $n_c = 2$, and $d_c = 2$ [units]. In the right case of Fig. 11, the green block in the k-th layer is placed on two 2×2 blocks with one-stud connection for each and on one 1×2 block with two-stud connection. Thus $n_b = 3$, $n_c = 4$ and $d_c = 2$ [units].

Larger n_b is preferable because the block placement on many blocks strengthens the scaffold structure, which may lead to changes of block assemblability (Case C and D). Larger n_c is preferable because the block placement on many studs leads to strong connection. We prefer smaller d_c so that blocks are placed from center to boundary, which reduces inter-block interference.

B. Block Prioritization

At first, the blocks to be placed in the k-th layer have Case-A, B, E or X connections. Here we place blocks whose connections are only Case-A or Case-B. Those blocks are prioritized according to the following criteria (in order of descending importance):

1) n_b is larger.
2) n_c is larger.
3) With more Case-A connections.
4) d_c is smaller.

The blocks are sequentially chosen for placement based on the priority.

After completing the above placement, we update the block assemblability. At this time, there may be the blocks with Case-C or Case-D connections. Now we prioritize assemblable blocks: that is, blocks without Case-X connections. The criteria is as follows (in order of descending importance):

1) With fewer Case-E connections.
2) With fewer Case-D connections.
3) With fewer Case-C connections.
4) n_b is larger.
5) n_c is larger.
6) With more Case-A connections.

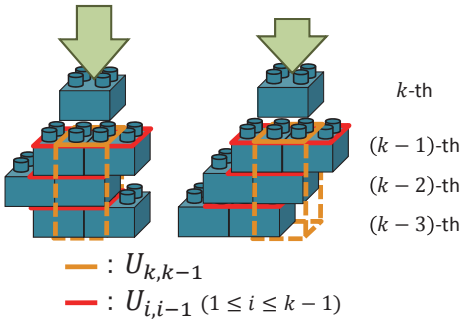


Fig. 12: Change Force According to Scaffolding Structure

7) d_c is smaller.

The block with the highest priority is chosen for the next placement. Then the priority is updated locally based on the placement. If all the blocks in the k-th layer have been chosen for placement, we proceed to the next layer. If there remain only blocks with Case-X connections, assembly of the k-th layer is infeasible.

If block placement orders for all the layers are successfully determined, assembly planning finishes.

VIII. Force Control in Assembly

In the process of block assembly, each block is inserted to the studs of the scaffold blocks. If the scaffold structure is strong enough, the block insertion force should be large to make the inter-block connection tight. However, in some cases, large insertion forces may cause collapse of the scaffold. Thus the applicability of such large forces must be judged and insertion force must be controlled appropriately for each block.

In this study, the applicability of large insertion forces is checked based on the contact region inclusion. Consider placing a block in the k-th layer. If the following equation holds for any i $(1 \le i \le k-1)$, we judge that large insertion force is applicable:

$$P(U_{k,k-1}) \subseteq P(U_{i,i-1}). \tag{4}$$

This is the case of the left of Fig. 12. Otherwise, as in the case of the right figure where (4) does not hold in the $(k-3)$-th layer, we judge that large insertion force is not applicable.

IX. Assembly Experiments

A. Experimental Setup

We prepared an experimental setup as shown in Fig. 13. A 6-axis industrial robot VP-6242G (Denso Wave) equipped with an electric gripper ESG1-F2840 (Taiyo) was used. We designed fingers with grooves to grasp studs of toy blocks (Fig. 14).

A Linux PC with Core i7 3930K (3.20 [GHz]) was used to control the robot according to generated assembly plans.

We used silicone block pads with 32×20 studs as a storage for building blocks. A plastic block plate with 20×20 studs was also used as a raft for assembly.

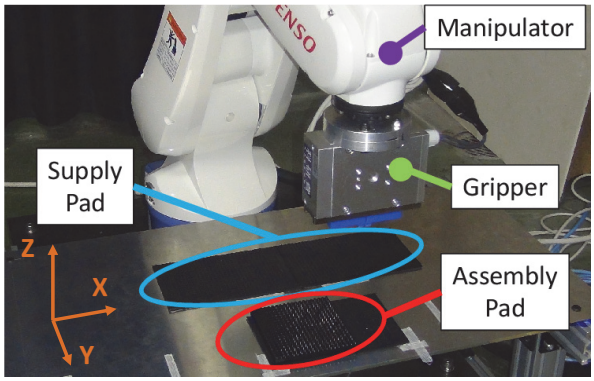


Fig. 13: Experimental Setup

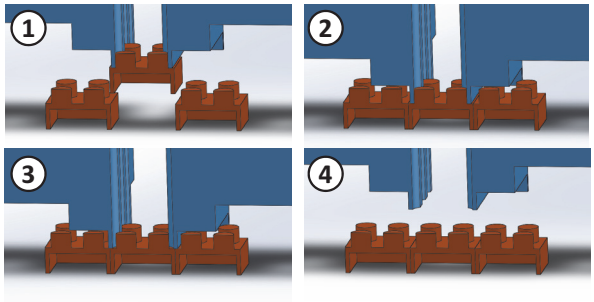


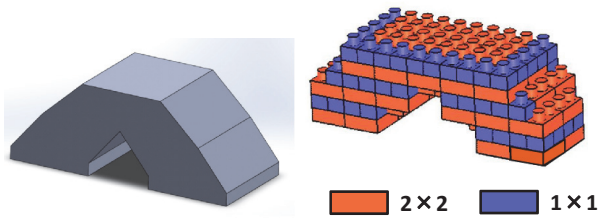
Fig. 14: Grasping Studs (Cross-sectional View)

B. Experimental Results

We tested our 3D printing system with the models shown in Fig. 15–18. The left CAD models were converted to the right block models as explained in Section IV. Table I shows the number of used blocks for each model.

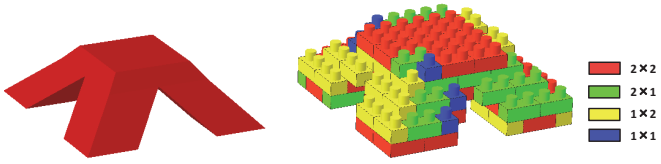
According to the block models, their assembly plans were generated, and executed by the robot. The robot has a force limitation function and it was disabled for the cases in which large insertion force was applicable and enabled for the others (see Section VIII). When enabled, the force in the vertical direction was limited to 60% of its maximum during the block insertion process.

Fig. 19 shows a scene in assembly execution. Fig. 20 shows the assembled brick sculptures. Note that the colors of the blocks in Fig. 19 and 20 are meaningless. All the models were successfully assembled. Table II shows time for assembly planning and execution. Compared to the execution time, the planning time is negligibly small.



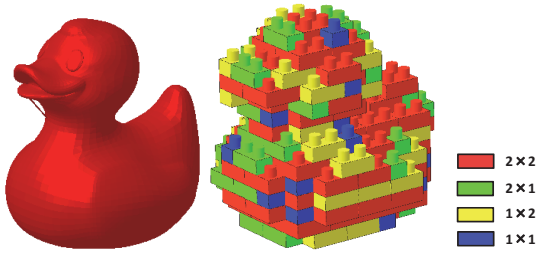
(a) CAD Model (b) Block Model

Fig. 15: An Arch



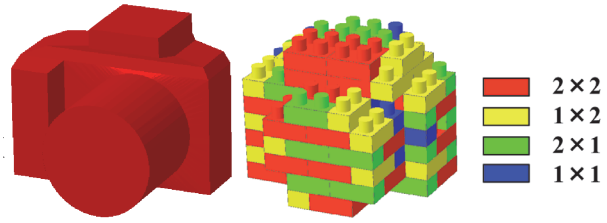
(a) CAD Model (b) Block Model

Fig. 16: A Quadpod



(a) CAD Model (b) Block Model

Fig. 17: A Duck



(a) CAD Model (b) Block Model

Fig. 18: A Camera

X. Conclusions

A robotic 3D block printer was presented. It can accept 3D CAD models and produce their corresponding brick sculptures. Several experiments demonstrated successful assemblies.

Future work should address making larger-scale models faster. Using a wider variety of primitive blocks including larger ones is a possible solution. Another problem is how to deal with CAD models that are unable to assemble. Such models can be divided into feasible subassemblies. Alternatively, we can add support blocks, which should be removed easily.

TABLE I: Number of Used Blocks

	Arch	Quadpod	Duck	Camera
1×1	60	7	17	9
1×2	0	43	64	40
2×2	58	54	110	46
Total	118	104	191	95

TABLE II: Time for Assembly

	Arch	Quadpod	Duck	Camera
Assembly Planning [s]	0.024	0.020	0.042	0.023
Assembly Execution [s]	880	790	1550	720

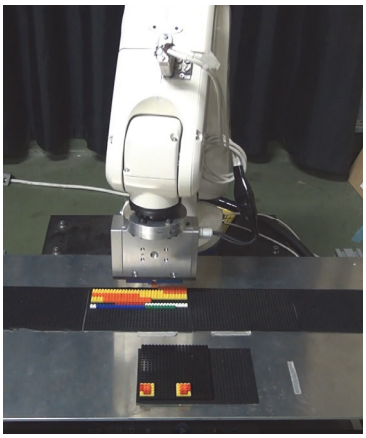


Fig. 19: A Scene in Arch Model Assembly

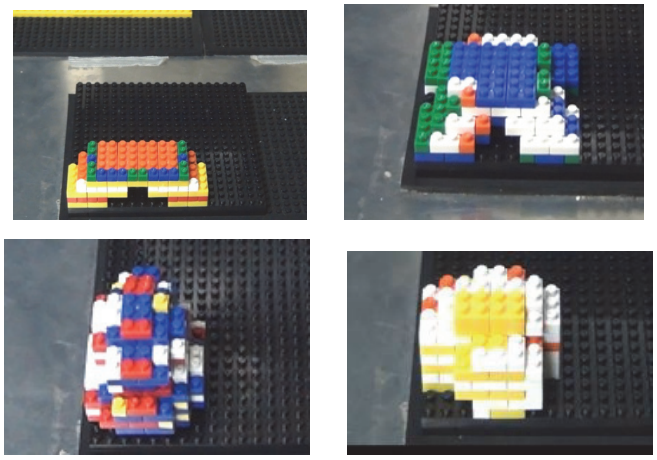


Fig. 20: Assembled Brick Sculptures

References

[1] S. H. Huang, P. Liu, A. Mokasdar, and L. Hou, "Additive manufacturing and its societal impact: a literature review," *Int. J. of Advanced Manufacturing Technology*, vol. 67, no. 5, pp. 1191–1203, 2013.

[2] S. Mueller, T. Mohr, K. Guenther, J. Frohnhofen, and P. Baudisch, "faBrickation: Fast 3D printing of functional objects by integrating construction kit building blocks," in *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, 2014, pp. 3827–3834.

[3] J. Hiller and H. Lipson, "Design and analysis of digital materials for physical 3D voxel printing," *Rapid Prototyping J.*, vol. 15, no. 2, pp. 137–149, 2009.

[4] "MakerLegoBot," http://www.battlebricks.com/makerlegobot/.

[5] J. D. Hiller and H. Lipson, "Fully recyclable multi-material printing," in *Solid Freeform Fabrication Proc.*, 2009, pp. 98–106.

[6] H. Medellin, J. Corney, J. Ritchie, and T. Lim, "Automatic generation of robot and manual assembly plans using octrees," *Assembly Automation*, vol. 30, no. 2, pp. 173–183, 2010.

[7] K. Sekijima, T. Masuda, and H. Tanaka, "Reconfigurable three-dimensional prototype system using digital materials," in *Trans. of Virtual Reality Soc. of Japan*, vol. 20, no. 2, 2015, pp. 97–105, (in Japanese).

[8] J. W. Kim, K. K. Kang, and J. H. Lee, "Survey on automated LEGO assembly construction," in *WSCG2014 Poster Paper Proc.*, 2014, pp. 89–96.

[9] T. Kozaki, H. Tedenuma, and T. Maekawa, "Automatic generation of LEGO building instructions from multiple photographic images of real objects," *Computer-Aided Design*, vol. 70, no. 1, pp. 13–22, 2016.