An Easy-to-Build Modular Robot Implementation of Chain-Based Physical Transformation for STEM Education

Minjing Yu[1], Ting Liu[2], Jeffrey Too Chuan Tan[3], and Yong-Jin Liu[4(✉)]

[1] College of Intelligence and Computing, Tianjin University, Tianjin 300350, China
minjingyu@tju.edu.cn
[2] Jupiter Robot Technology Co., Ltd., JiaXing 314000, China
lucia@jupiterobot.com
[3] MyEdu AI Robotics Research Centre, Selangor 47301, Malaysia
i@jeffreytan.org
[4] Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
liuyongjin@tsinghua.edu.cn

Abstract. The physically realizable transformation between multiple 3D objects has attracted considerable attention recently since it has numerous potential applications in a variety of industries. In this paper, we presented EasySRRobot, a low-cost, easy-to-build self-reconfigurable modular robot, to realize the automatic transformation across different configurations, and overcomes the limitation of existing transformation methods requiring manual involvement. All on-board components in EasySRRobot are off-the-shelf and all support structures and shells are 3D printed, so that any novice users can make it at home. In addition, an algorithm to automatically find an optimal design for the interior structure was proposed, and the result has been demonstrated by comparing with another two feasible designs. Thirty modules were fabricated with the aid of 3D printing and the motions of two configurations (snake and wheel) were realized, which shows the working ability and effectiveness of the proposed EasySRRobot. We further explored the effect of EasySRRobot on spatial ability, a skill that is crucial for STEM education. The results indicated that interacting with EasySRRobot can effectively improve the performance of the transformation task, suggesting that it might improve mental rotation skills and other aspects of spatial ability.

Keywords: Self-reconfigurable · Modular Robots · Transformation · Spatial Ability

1 Introduction

The physically realizable transformation between multiple 3D objects has been utilized extensively in aerospace, education, entertainment, and other industries

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024
S.-M. Hu et al. (Eds.): CAD/Graphics 2023, LNCS 14250, pp. 170–185, 2024.
https://doi.org/10.1007/978-981-99-9666-7_12

due to the quick growth of computer and material science. Compared to models that can be disassembled, the transformation between those non-detachable 3D models is more challenging. All modules of them are connected and the configurations can only be transformed by shifting, folding, and twisting [6,11,18]. However, the transformation process for such studies requires manual human involvement, which is laborious and time-consuming. With the development of robotics, self-reconfigurable modular robots have made automatic transformation possible.

A self-reconfigurable modular robot (SRRobot) is capable of altering their shapes and functionalities when the task or environment is changed. It is constructed from modules, each of which is physically independent. There are many types of SRRobot units, such as double-cube modules. Note that all double-cube modules have similar exterior structures and similar functions. However, these modules can have diverse interior structures for placing on-board components, such as actuators, sensors, batteries, microprocessors and intermodule communication/power-transmission/connection devices, etc. In this paper, we present a low-cost and easy-to-build SRRobot, called *EasySRRobot*, to realize automatic transformation between models. All on-board components in EasySRRobot, are off-the-shelf and all support structures and shells are 3D printed, so that any novice users can make it at home. A distinct feature of EasySRRobot is that it uses an optimized interior structure design. An algorithm has been proposed in this paper to automatically find an optimal interior structure design for placing a given set of on-board components in a double-cube module. To demonstrate the effectiveness and usefulness of our algorithm, we compare the optimal design output from our algorithm with another two feasible designs and build a EasySRRobot prototype using the optimal design.

Furthermore, we explored the impact of chain-based physical transformation using EasySRRobot on STEM (Science, Technology, Engineering and Mathematics) education, and we focus on spatial ability, which is a category of human reasoning skills that plays an important role in affecting a person's development in STEM. Spatial ability has been demonstrated to be malleable and can be improved through training. In this paper, we present a training scheme by tangible interaction with EasySRRobot, and a preliminary user study based on behavioral and EEG data analysis shows that via interaction with EasySRRobot, users can significantly improve their performance on a task related to spatial ability.

2 Related Work

Self-reconfigurable Modular Robot. An SRRobot consists of modules, each of which is physically independent and encapsulate a certain simple function. Complex tasks performed by SRRobots are realized by the joint function of modules. An SRRobot can change their shape and functionality when the task or environment is changed. Usually the modules in an SRRobot are identical and then self-repair is easy for SRRobots by simply replacing broken modules with

good ones. Due to these nice properties, SRRobots have attracted considerable attention recently. Many types of SRRobots have been proposed, among which the ones with double-cube modules have been widely used, including M-TRAN series [8,9,13], SuperBot [14] and Dtto [5], etc. Most of these double-cube module are constructed with professional electronic/magnetic components. Fabricating and assembling the cubes with these on-board components need professional equipments, production process and skills, which are only available at professional labs or factories.

Spatial Ability. Spatial ability is a category of human capacity to understand, reason and remember the spatial relations among objects, which makes use of basic memory for shape and position [2]. Spatial ability of children or teenagers is highly correlated with their achievement in advanced science, technology, engineering and mathematics (STEM) [17]. Previous studies [15] showed that spatial ability can be improved by training. The training program for spatial skills has been divided into three mutually exclusive categories [16]. The first category, which is frequently carried out in a laboratory setting, instructs the participants on spatial tasks by specialized practice, rehearsal, or reading an instruction manual. The second category of training involves providing courses that concentrate on the improvement of spatial skills, and the training period can range from weeks to a year. However, it has been suggested that these two traditional spatial training methods may not cover all aspects of spatial skills that are used when humans work in 3-D physical space [7]. As consequently, there is an urgent demand for a better training category that utilized active exploration of the real physical environment.

3 Hardware and Module Design Principle

In this section, we briefly summarize the design principles that are common in most double-cube modules and present our specified hardware that we used in EasySRRobot. The reason of choosing our specified on-board components is that these components and their drivers are off-the-shelf and their costs are low, so that any novice users can easily make the EasySRRobot at home. A double-cube module consists of two boxes and a link (Fig. 1), and it has two degrees of freedom. Both boxes have identical shape and we choose the semi-cylindrical box as the module's shape in EasySRRobot. We follow the M-TRAN III [9] to use a mechanical connection in EasySRRobot. The connection mechanism makes the two boxes in a module differ in gender (Fig. 1):

Male box: slits exist on its three planar faces, such that hooks can be rotated out to latch a female box in another module;

Female box: hook cavities exist on its three planar faces, such that hooks from male boxes can latch into the faces.

3.1 Hardware

Two HX1218D servomotors are used to rotate the male and female boxes, respectively. Three SG90 servomotors are used to drive the hooks in the male box.

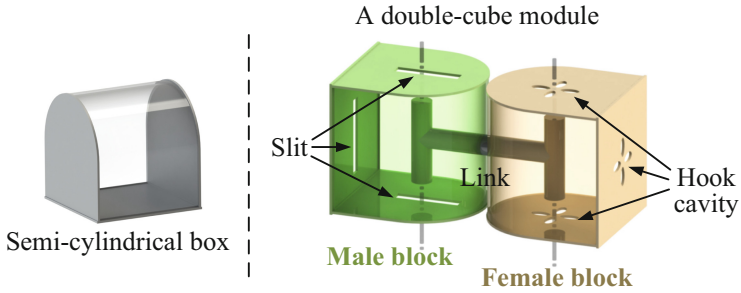


Fig. 1. A double-cube module consists of two boxes and a link. Both boxes have identical shape and each of them can rotate around its axis by $\pm 90°$.

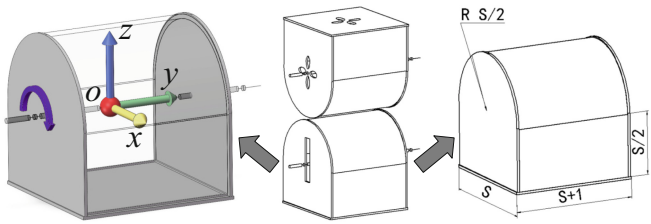


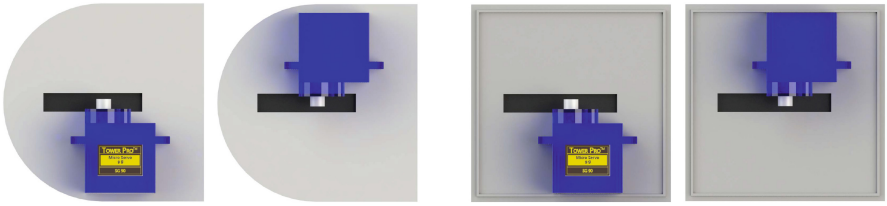
Fig. 2. The coordinate system $\{o, x, y, z\}$ (left) of the double-cube module (middle) and dimensions of the male box (right).

These five servomotors are driven by a control circuit built on the Arduino MCU with ATmega328P CPU. The control circuit also integrates a HC-05 bluetooth module and an nRF24L01 transceiver IC. A double-cube module can communicate with a host PC via the bluetooth module and any two modules can communicate with each other via the transceiver IC. The power to the circuit is supplied by a Li-Po7.4v 500 mAh battery. All these components and their drivers are off-the-shelf.

3.2 Spatial Relationships and Design Principle

To place on-board components into a double-cube module, design constraints reflecting spatial relationships need to be established. To specify these constraints, we first setup a coordinate system in the module (Fig. 2): the origin o is at the center of the male box, the rotation axis of the male box is along the y direction, and the center of female box is in the $+z$ direction.

Both male and female boxes in the module have the same shape, and we characterize the size of male box using parameterized dimensions S, $S+1$ and $S/2$ in x, y and z directions (Fig. 2 right), where the integer parameter S will be optimized by the algorithm presented in the next section. Let $\Omega_M(s)$ and $\Omega_F(s)$ be the spaces enclosed by the boundary surfaces of male and female boxes, respectively. The first design constraint is:



(a) Semi-disk face (b) Rectangular face

Fig. 3. A SG90 servomotor c_{SG_i} has only two candidate locations $\{\Theta_{SG_i,1}, \Theta_{SG_i,2}\}$ on one of faces with slits in the male box.

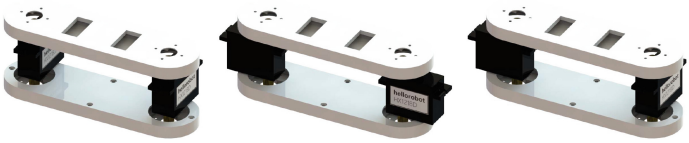


Fig. 4. Three possible orientations of two HX1218D servomotors with the link.

Constraint 1. The axis-aligned bounding box of each on-board component is inside $\Omega_M(S) \cup \Omega_F(S)$.

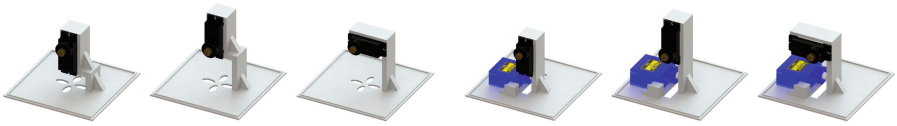
For each on-board component c_i, denote its axis-aligned bounding box as $BBox(c_i)$. In the six faces of $BBox(c_i)$, we choose one of two faces that have the maximal area as the base face. Then any location of c_i can be specified by $\Theta_i = (x_i, y_i, z_i, \alpha_i, \beta_i)$, where (x_i, y_i, z_i) is the position of the center of $BBox(c_i)$ and (α_i, β_i) are two spherical angles indicating the normal of the base face. In the following, we specify the hierarchical spatial relationships among on-board components.

The three SG90 servomotors $\{c_{SG_1}, c_{SG_2}, c_{SG_3}\}$ drive the hooks in the male box. Then each of them should be placed on a face with a slit in the male box. Noting that each slit is in the middle of the face (see Fig. 3), the second design constraint is

Constraint 2. each of three SG90 servomotors can only have two candidate locations $\Gamma_{SG_i} = \{\Theta_{SG_i,1}, \Theta_{SG_i,2}\}$, $i = 1, 2, 3$.

The two HX1218D servomotors $\{c_{HX_1}, c_{HX_2}\}$ drive the link in the module. To realize the rotation of each box around its axis, each box has a HX1218D servomotor and the rotation axis of this servomotor coincides with the rotation axis of the box (i.e., in the y direction). Moreover, the line connecting the centers o_{HX_1} and o_{HX_2} of two servomotors should be perpendicular to the y direction, such that a link (see Fig. 4) can be placed with these two servomotors. To summarize:

Constraint 3. All the possible locations of a HX1218D servomotor c_{HX_1} in the male box can be characterized by a subset of $\mathbb{R}^2 = (y_{HX_1}, \alpha_{HX_1})$ denoted as



(a) Three examples of supporting structures in the female box

(b) Three examples of supporting structures in the male box

Fig. 5. Some examples of supporting structures of HX1218D servomotors for the orientations shown in Fig. 4.

Γ_{HX_1}, because x_{HX_1}, z_{HX_1} and β_{HX_1} are fixed to guarantee that the rotation axis of the servomotor is in the y direction.

Constraint 4. All the possible locations of another servomotor c_{HX_2} in the female box can be characterized by a subset of $\mathbb{R} = (\alpha_{HX_2})$ denoted as Γ_{HX_2}, due to the constraint $y_{HX_2} = y_{HX_1}$.

Since the HX1218D servomotor has two ends along its rotation axis that can mount the link, in our setting, the link consists of two disjoint components as shown in Fig. 4.

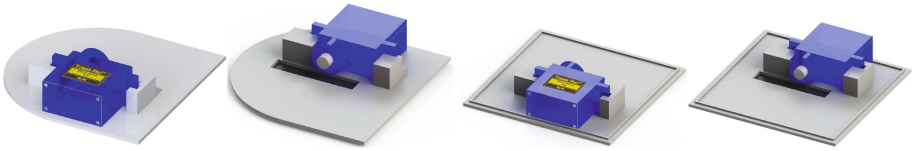
The control circuit has two pieces $\{c_{ctrl_1}, c_{ctrl_2}\}$. Denote the sets of all the possible locations of c_{ctrl_1}, c_{ctrl_2} and $c_{battery}$ as Γ_{ctrl_1}, Γ_{ctrl_2} and $\Gamma_{battery}$ respectively, which are subsets of $\mathbb{R}^5 = (x_i, y_i, z_i, \alpha_i, \beta_i)$, $i = ctrl_1, ctrl_2, battery$. Then Constraint 1 can be re-expressed as

Constraint 5. The locations of control circuit and battery are in Γ_{ctrl_1}, Γ_{ctrl_2} and $\Gamma_{battery}$ respectively.

4 The Optimization Algorithm

In this section, we present an algorithm that automatically optimizes the interior structure design of the double-cube module in EasySRRobot.

First, we choose four candidate values S = 68 mm, 72 mm, 76 mm, 81 mm that will be optimized as the parameterized dimensions of the boxes (Fig. 2 right). Second, for each on-board component c_i, once its location is specified, our algorithm automatically add a supporting structure to fix it. See Fig. 5 and 6 for the schemes of adding supporting structures of five servomotors. Third, note that each c_i of on-board components has its restricted location space Γ_i, $i = SG_1$, SG_2, SG_3, HX_1, HX_2, $ctrl_1$, $ctrl_2$ and $battery$. Denote the Cartesian product of the topological spaces Γ_i as $\Gamma = \prod_i \Gamma_i$. Then, any feasible placement of all on-board components corresponds to a point $p \in \Gamma$. Note that once the locations of c_{HX_1} and c_{HX_2} are determined, the two link components are determined. Fourth, we propose an objective function that evaluates the fitness of any collision-free placement $p \in \Gamma$ based on three evaluation criteria. Finally, the optimal interior structure design is obtained by minimizing the objective function in the search space Γ using the simulated annealing method.



(a) Semi-disk face (b) Rectangular face

Fig. 6. Supporting structures of SG90 servomotors for each of two candidate locations shown in Fig. 3.

4.1 Objective Function

The objective function F consists of three terms, i.e., structural soundness, space utilization and assembly complexity:

$$F = w_1 f_{struct} + w_2 f_{space} + w_3 f_{assembly} \tag{1}$$

where w_1, w_2 and w_3 are non-negative weights.

Structural Soundness. We perform a structural analysis on this feasible interior structure using the library *SfePy* [4], by applying gravity and torques from servomotors as the external force. Let σ_{abs_max} be the maximal absolute stress in this structure. Then

$$f_{struct} = \frac{\sigma_{abs\_max}}{\sigma_{ref}} \tag{2}$$

where σ_{ref} is a reference stress, which is computed as an average of maximal absolute stresses on five manually generated interior structures. The smaller f_{struct}, the better structural soundness.

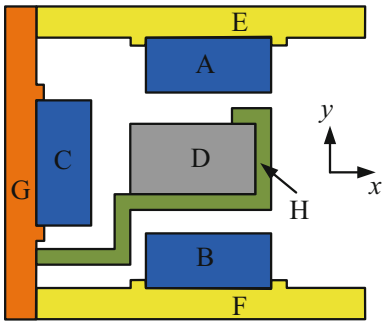
Space Utilization. A good design should take full advantage of the box space. The space utilization ratio f_{space} is defined to be the ratio of the volume of all on-board components to the box space in the double-cube module, i.e.,

$$f_{space} = \frac{Vol_{male} + Vol_{female}}{\sum_{i=1}^{8} Vol(c_i)} \tag{3}$$

where $Vol(c_i)$ is the volume of component c_i, Vol_{male} and Vol_{female} are volume of male and female boxes, respectively. In some literatures (e.g., [9]), the power-weight ratio is used. Given that the power of servomotors is fixed, the power-weight ratio is closely related to the space utilization f_{space}. The smaller f_{space}, the better space utilization.

Assembly Complexity. A collision-free placement of all on-board components does not mean that they can be assembled validly. Even if so, different assembly sequences can have different assembly complexities. A good structural design should have a low assembly complexity, i.e., easy for assembly and disassembly.

To automatically check the validity of a assembly sequence and compute the assembly complexity, we use an *assembly matrix* representation M [3] to



Assembly matrix

	A	B	C	D	E	F	G	H
A	\	±x−y	±x±y	±x−y	+y	±x−y	−x±y	±x−y
B	±x+y	\	±x±y	±x+y	±x+y	−y	−x±y	−x+y
C	±x±y	±x±y	\	+x±y	±x+y	±x−y	−x	+x−y
D	±x+y	±x−y	−x±y	\	±x+y	±x−y	−x±y	+x
E	−y	±x−y	±x−y	±x−y	\	±x−y	−x−y	±x−y
F	±x+y	+y	±x+y	±x+y	±x+y	\	−x+y	±x+y
G	+x±y	+x±y	+x	+x±y	+x+y	+x−y	\	+x−y
H	±x+y	+x−y	−x+y	−x	±x+y	±x−y	−x+y	\

Fig. 7. A 2D assembly structure (left) and its assembly matrix M (right).

describe the geometric constraints between components in an assembly. M is a square $n \times n$ matrix, where n is the number of components in the assembly. The (i, j) entry of M stores directions with which the component c_i can be assembled without colliding with the component c_j. A simple 2D example is illustrated in Fig. 7, which has eight components. For instance, at the entry $(1, 2) = (A, B)$, the directions $\pm x - y$ means that A can be assembled along $\pm x$ or $-y$ directions, without colliding with B.

Given an assembly sequence $\prod = (c_1, c_2, \cdots, c_n)$, we compute $V(i) = \bigcap_{j<i} M(i, j)$ for each c_i. Noting that c_j, $j < i$, is a component assembled before c_i, $V(i)$ gives valid assembly directions for c_i. $\prod$ is a valid assembly sequence, if and only if $V(i) \neq \emptyset$, $\forall i$. For example, in the 2D case in Fig. 7, if $\prod = (G, C, H, D, A, B, E, F)$, we have $V(D) = (-x, \pm y) \bigcap (-x, \pm y) \bigcap (+x) = \emptyset$. Then $\prod$ is not a valid assembly sequence.

Given a valid assembly sequence $\prod = (c_1, c_2, \cdots, c_n)$, we define its assembly complexity as the number of re-orientations, which can be readily computed from $V(i)$:

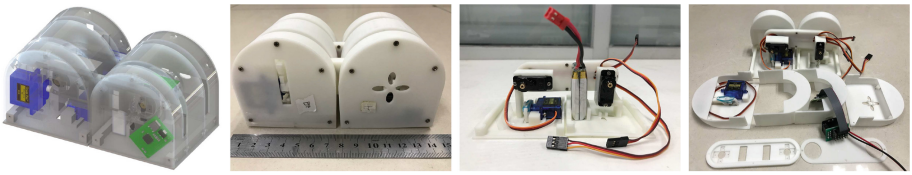
if $\bigcap_{k=i}^{j} V(k) \neq \emptyset$, no re-orientation is needed during the assembly of $c_i, \cdots, c_j$;

if $\bigcap_{k=i}^{j} V(k) \neq \emptyset$ and $\bigcap_{k=i}^{j+1} V(k) = \emptyset$, one re-orientation is needed for assembling c_{j+1}.

For example, $\prod = (G, C, A, E, D, H, B, F)$ is a valid assembly sequence in the 2D case in Fig. 7, which needs two re-orientations. Given a collision-free placement $p \in \Gamma$, we use the generic algorithm [3] to compute an optimal assembly sequence $\prod$ and set the assembly complexity metric as

$$f_{assembly} = \begin{cases} 10^6 & \text{if } \prod \text{ is not valid} \\ n_{reorient} & \text{otherwise} \end{cases} \tag{4}$$

where $n_{reorient}$ is the number of re-orientations in a valid $\prod$.



(a) Rendering model (b) Real model (c) Two intermediate statuses in the assembly process

Fig. 8. The optimal design result output from our proposed algorithm. (a) is a rendering of 3D CAD model with shells being semi-transparency. (b) is a real assembled model by 3D printing. (c) shows two intermediate statuses in the assembly process.

4.2 Simulated Annealing

Finding an optimal solution to minimizing the objective function (1) in the search space $\Gamma = \prod_i \Gamma_i$ is a mixed combinatorial and continuous optimization problem, because the subspaces Γ_i, $i = SG_1, SG_2, SG_3$, are discrete and other subspaces are continuous.

Simulated annealing (SA) [10] is probabilistic technique that can efficiently approximate the global optimum. To apply the SA technique, we first generate a random solution s in Γ and compute its cost $F(s)$ using the objective function (1). We say a point $p \in \Gamma$ is a solution s if p corresponds to a collision-free placement of all the on-board components. Then we generate a random neighboring solution s' and compute the cost $F(s')$. The Metropolis acceptance criterion is adopted to determine how the system moves from the current solution s to the candidate solution s', with the acceptance probability p:

$$p = \begin{cases} e^{-\frac{F(s')-F(s)}{T}} & \text{if } F(s') - F(s) > 0 \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

where T is the temperature. It was shown [10] that if the temperature was cooled slowly, the global minimum can be found. The system iteratively updates the solution until a specified iteration number is reached and the cost does not decrease anymore.

5 Implementation

We implement the proposed algorithm in C++ and test it on a PC with an Intel E5-2650 CPU (2.60 GHz) and 64 GB RAM. Since structural soundness is more important than space utilization and assembly complexity, in all our experiments, we set $w_1 = 10.0$, $w_2 = 1.0$ and $w_3 = 1.0$ in the objective function 1. The algorithm takes 24 min to compute the optimal design result, which is illustrated in Fig. 8(a). Thirty modules were manufactured for EasySRRobot with the aid of 3D printing (Fig. 8(b)). Since all on-board components are off-the-shelf and assembly complexity is considered in the algorithm, assembling these components into modules of EasySRRobot is easy (see Fig. 8 (c)).

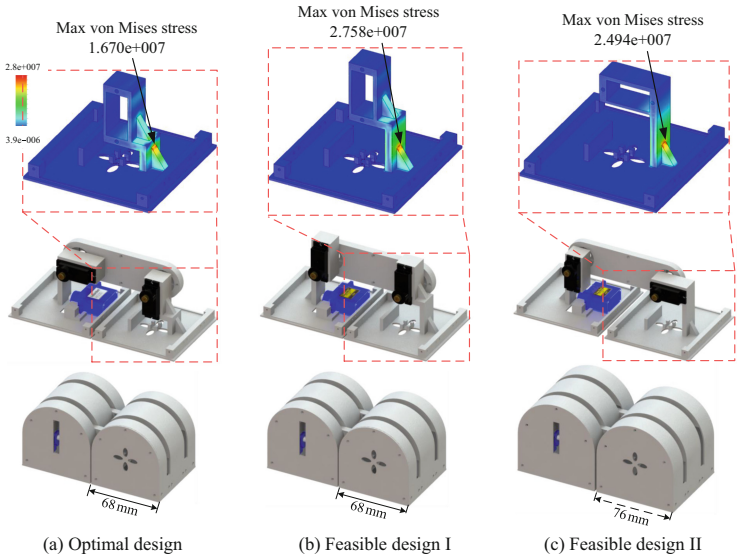
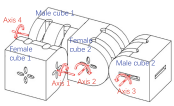


Fig. 9. The comparison of our optimal design (a) with two feasible designs (b) and (c).

To show the advantage of the proposed algorithm, we compare the optimal design determined by our algorithm with two feasible designs. In our optimal design (Fig. 9(a)), the orientations of two HX1218D servomotors (shown in black texture) are perpendicular. In the feasible design I (Fig. 9(b)), the orientations of two HX1218D servomotors (black) are the same. Although these two designs have the same box size, the maximum stress ($2.758 \times 10^7 \mathrm{N/m}^2$) of the feasible design I is larger than the one ($1.670 \times 10^7 \mathrm{N/m}^2$) in the optimal design, resulting in a larger value of the structural soundness metric (ref. Eq. (2)). In the feasible design II (Fig. 9(c)), the orientations of two HX1218D servomotors (black) are also perpendicular. Due to the potential collision between the HX1218D servomotor (black) and a SG90 servomotor (blue) in the male box, the box size of the feasible design II has to be larger ($S = 76$ mm) than the one ($S = 68$ mm) in the optimal design, resulting in a larger value of the space utilization metric (ref. Eq. (3)). Meanwhile, the maximum stress ($2.494 \times 10^7 \mathrm{N/m}^2$) of the feasible design II is also larger than the one ($1.670 \times 10^7 \mathrm{N/m}^2$) in the optimal design.

We further examine the optimal design in terms of the self-reconfiguration functionality. We build a EasySRRobot prototype using the manufactured modules. Our algorithm has good versatility and scalability. If other types of components are provided, our algorithm could also calculate the corresponding optimized design solution if the appropriate constraints are given.



(a) Two connected modules

(b) Physical prototype

(c) Rotation 1

(d) Rotation 2

(e) Rotation 3

(f) Rotation 4

(g) Rotation 5

(h) Rotation 6

(i) Rotation 7

(j) Rotation 8

(k) Rotation 9

Fig. 10. Two connected modules can support totally $3 \times 4 \times 4 = 48$ rotations and nine rotations are illustrated here.

6 User Study

We studied the role of EasySRRobot in improving spatial ability using two module settings: an EasySRRobot with one module and with two connected modules.

The rotation DOFs provided by these two settings are as follows. Given one module, each of two cubes can rotate around its own axis or the axis of the other cube, and therefore, can support $2 \times 2 = 4$ types of rotations. Two modules can be connected in three different ways, according to the faces between which the connection is established. Given that each module supports four different rotations, users can interact with two modules by realizing totally $3 \times 4 \times 4 = 48$ rotations (Fig. 10). Users can interact with EasySRRobot in two ways: (1) The robot changes shape autonomously according to different levels of tasks (easy or difficult) and a user observes the transformation process; (2) A user holds the cubes and interactively rotate them.

The Purdue visualization of rotations (ROT) test [1] has been used in our work, which is one of the most commonly used measures of spatial ability and is suitable for testing individuals who are at least thirteen years old.

Transformation Task Description. Participants were presented with pairs of drawings of EasySRRobot (Fig. 11). In each pair, the upper item shows a reconfiguration of EasySRRobot with given rotations, i.e., an original shape of an EasySRRobot is reconfigured into a target shape and this transformation is achieved by applying given rotation operations. The lower item is the question to be answered, i.e., by changing the original shape and applying the same rotations as in the upper item, the participants were asked to choose the correct target shape. There were 20 successive tasks (10 easy and 10 difficult). Easy level only involved one module and difficult level involved two connected modules. Tasks were presented at the center of a 27-inch LED screen with a recommended 1680×1050 pixel resolution. The screen was located 60cm from the participants'

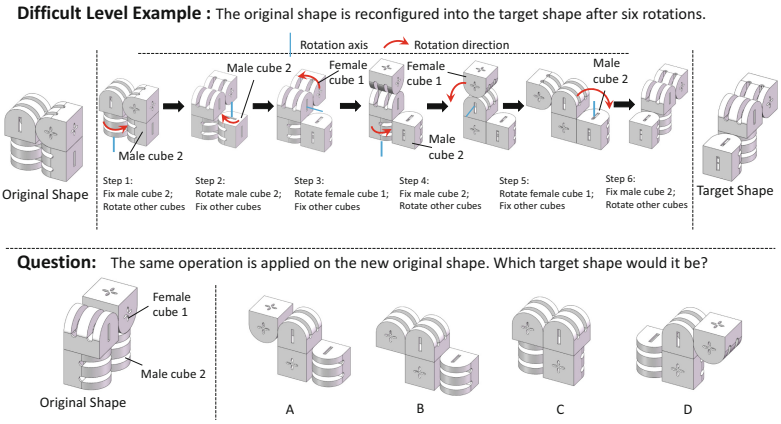


Fig. 11. An example of the task at difficult level (two connected modules).

eyes. All participants were instructed to complete the tasks as quickly as possible on the premise of ensuring the correct rate.

Participants. 18 undergraduate and graduate students (9 males and 9 females) were selected to take part in this user study. Their ages ranged from 19 to 34 years old ($average = 24.33$, $SD = 3.93$).

Experimental Procedure. Upon arrival, each participant put on an EEG cap with the assistance of two experimenters. Prior to the formal test, each participant went through a ROT test for evaluating his/her spatial ability. Then participants were partitioned into two groups (namely experimental group and control group) and the ROT test results of two groups were ensured at the same level. We used a 2×2 mixed design with group as a between-subject variable (i.e., experimental group and control group are with and without training by interaction with EasySRRobot respectively) and testing session as a within-subject variable (pre-test vs. post-test). Each participant completed three consecutive sessions: a transformation task (i.e., pre-test), training session, and another transformation task (i.e., post-test). The pre-test and post-test were the same for two groups of participants. All participants were instructed to complete 10 successive transformation tasks (5 easy and 5 difficult) for examining the effect of training. To avoid a potential confounding effect, the operation order was counterbalanced in both pre- and post-test. During the training session, the experimental group was required to interact with EasySRRobot for familiarizing themselves with the rotation rules in EasySRRobot, while the control group was required to learn by reading a printed material.

EEG Acquisition. EEG data were continuously recorded from 64 active electrodes attached to an electrode elastic cap (Neuroscan Inc., Charlotte, NC). Electrode positions included the standard International 10–20 system locations and intermediate sites. The left mastoid was used as an online reference for all

Table 1. Mean and standard deviation (in bracket) of normalized alpha power of EEG signals at 10 electrode sites.

Site	Before Training		After Training	
	Experimental Group	Control Group	Experimental Group	Control Group
F3	0.487(0.272)	0.546(0.322)	0.441(0.261)	0.560(0.360)
F4	0.527(0.298)	0.551(0.341)	0.573(0.206)	0.654(0.244)
Fz	0.478(0.275)	0.527(0.364)	0.400(0.263)	0.570(0.373)
F7	0.527(0.309)	0.560(0.281)	0.446(0.135)	0.582(0.320)
F8	0.488(0.260)	0.656(0.297)	0.405(0.260)	0.697(0.298)
FCz	0.452(0.250)	0.468(0.358)	0.456(0.237)	0.511(0.365)
Cz	0.432(0.271)	0.467(0.257)	0.453(0.245)	0.535(0.293)
P3	0.489(0.246)	0.510(0.246)	0.471(0.276)	0.595(0.253)
P4	0.401(0.275)	0.539(0.287)	0.406(0.248)	0.560(0.314)
Pz	0.474(0.240)	0.569(0.225)	0.452(0.232)	0.622(0.267)

channels. The EEG data were digitized at 500 Hz. The alpha power (8–12 Hz) spectral features of the EEG signals on 10 channels were extracted and the results were summarized in Table 1. These 10 channels located in the frontal (F3, F4, Fz, F7, F8), midline (FCz, Cz), and parietal (P3, Pz, P4) areas. The power values when experiencing the transformation tasks minus the power values of the resting period, which was finally normalized to the range [0,1].

Results. We first examined the difference in the accuracy of ROT test to ensure that the spatial ability of two groups was at the same level before completing the transformation task. There was no significant difference in the accuracy of ROT test for mental rotation between two groups. The average accuracy was 0.83 ($SD = 0.11$) for the participants in the experimental group, while the average accuracy was 0.79 ($SD = 0.13$) for those in the control group.

Then we analyzed the differences in the transformation task performance between the pre-test and post-test sessions and between two groups. We found that the experimental group achieved better performance on the transformation task by interaction with EasySRRobot. Three behavioral responses, *time to completion* (TTC), *time to correct completion* (TTCorrect) and correct rate, were recorded:

- TTC measures the amount of average time (in seconds) that a participant spends to complete a transformation task.
- TTCorrect measures the amount of average time (in seconds) that a participant spends to correctly complete a transformation task.
- Correct rate measures the ratio of correct answers in all tasks.

We compared the training effects on TTC and TTCorrect between experimental and control groups. There was a significant *group* × *testing session* interaction for the TTCorrect ($F(1, 16) = 7.09$, $p = 0.017$) and a marginal

significant *group* × *testing session* interaction for the TTC ($F(1, 16) = 3.65$, $p = 0.074$). The simple effect analysis showed that participants in the experimental group spent less TTCorrect time ($M = 93.09$, $SD = 29.18$) after the training through interacting with EasySRRobot, resulting in a 30.76% improvement in the TTCorrect. As a comparison, after training through reading printed material, in the control group the improvement in TTCorrect was only 1.54%. Neither the main effect of group nor the main effect of testing session was significant for the TTCorrect. The simple effect analysis of the *group*×*testing session* interaction showed the similar pattern for the TTC. Moreover, there was no significant difference in the correct rate between two groups or between two testing sessions.

In addition to behavioral indices, EEG correlates of mental rotation were also analyzed to examine the neural mechanism underlying the transformation task. Previous study revealed that the suppression of alpha power increased with the task difficulty [12]. We found that when the experimental group completed the transformation task in the post-test session, the alpha power of EEG signals was significantly suppressed, indicating that the experimental group may invest more cognitive resources in the task related to spatial ability after the training process. Specifically, There was a significant *group* × *testing session* interaction for the mean alpha power at the Fz electrode size ($F(1, 16) = 5.11$, $p = 0.038$). In particular, the mean alpha power decreased from 0.48 to 0.40, leading to a 16.48% suppression of the alpha activity. As a comparison, the mean alpha power remained at the similar level when the participants in the control group were engaged in the transformation task, after training through reading paper material. Neither the main effect of group nor the main effect of testing session was significant for the mean alpha power at the Fz electrode size. There was no significant difference in the mean alpha power in the midline or parietal areas.

Given the findings at the Fz electrode size, we continued to explore the asymmetric brain activation in the frontal areas. EEG asymmetric features were calculated by subtracting the mean alpha power values in the left hemisphere from the mean alpha power values in the corresponding right hemisphere (e.g., F3-F4, F7-F8). We observed more brain activation in the left frontal area after the training process for both of groups. Specifically, there was a significant main effect of the testing session for the mean alpha power asymmetry at the pair of F3-F4 ($F(1, 16) = 6.99$, $p = 0.018$). After training, the mean alpha power in the left frontal area was significantly less than the value in the right frontal area when participants were engaged in the transformation task. No significant difference in this measure was found between two groups. The mean alpha power asymmetry at another pair of F7-F8 was not significant between two groups or between two testing sessions.

7 Conclusion and Future Work

In this paper, we present a low-cost and easy-to-build self-reconfigurable modular robot called EasySRRobot, which enabled automatic chain-based physical transformation. And we investigated the effect of EasySRRobot on the enhancement

of spatial ability, a skill that is important for STEM learning. Results on both reaction time and accuracy indicated that training by interaction with EasySRRobot can effectively improve the performance of the transformation task, which means training through interaction with EasySRRobot might improve mental rotation skills and other aspects of spatial ability. In this work, we propose a new training scheme by using active, physical exploration of the real world through tangible interaction. According to our current results, we expect that the current training scheme can improve not only mental rotation skills but also other aspects of spatial ability. Our future investigation will continue to work along this research line.

References

1. Bodner, G.M., Guay, R.B.: The Purdue visualization of rotations test. Chem. Educ. **2**(4), 1–17 (1997)
2. Carpenter, P.A., Just, M.A.: Spatial ability: an information processing approach to psychometrics. In: Advances in the Psychology of Human Intelligence, vol. 3. Hillsdale, NJ: Lawrence Erlbaum Associates (1986)
3. Chen, S.F., Liu, Y.J.: An adaptive genetric assembly-sequence planner. Int. J. Comput. Integr. Manuf. **14**(5), 489–500 (2001)
4. Cimrman, R., Lukeš, V., Rohan, E.: Multiscale finite element calculations in python using SfePy. Adv. Comput. Math. **4**, 1897–1921 (2019)
5. Dtto_Modular_Robot_V2.0 (2016). https://hackaday.io/project/9976-dtto-v20-modular-robot
6. Garg, A., Jacobson, A., Grinspun, E.: Computational design of reconfigurables. ACM Trans. Graphics (TOG) **35**(4), 90:1–90:14 (2016)
7. Kaufmann, H., Steinbügl, K., Dünser, A., Glück, J.: General training of spatial abilities by geometry education in augmented reality. Ann. Rev. CyberTherapy Telemed. Decade VR **3**, 65–76 (2005)
8. Kurokawa, H., Kamimura, A., Yoshida, E., Tomita, K., Kokaji, S., Murata, S.: M-TRAN II: metamorphosis from a four-legged walker to a caterpillar. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '03), pp. 2454–2459 (2003)
9. Kurokawa, H., Tomita, K., Kamimura, A., Kokaji, S., Hasuo, T., Murata, S.: Distributed self-reconfiguration of M-TRAN III modular robotic system. Int. J. Robot. Res. **27**(3–4), 373–386 (2008)
10. Laarhoven, P.J.M., Aarts, E.H.L. (eds.): Simulated Annealing: Theory and Applications. Kluwer Academic Publishers, Norwell, MA, USA (1987)
11. Li, H., Hu, R., Alhashim, I., Zhang, H.: Foldabilizing furniture. ACM Trans. Graph. (TOG) **34**(4), 90:1–90:12 (2015)
12. Michel, C.M., Kaufman, L., Williamson, S.J.: Duration of EEG and MEG α suppression increases with angle in a mental rotation task. J. Cogn. Neurosci. **6**(2), 139–150 (1994)
13. Murata, S., Yoshida, E., Kamimura, A., Kurokawa, H., Tomita, K., Kokaji, S.: M-TRAN: self-reconfigurable modular robotic system. IEEE/ASME Trans. Mechatron. **7**(4), 431–441 (2002)
14. Salemi, B., Moll, M., Shen, W.: SUPERBOT: a deployable, multi-functional, and modular self-reconfigurable robotic system. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '06), pp. 3636–3641 (2006)

15. Uttal, D.H., et al.: The malleability of spatial skills: a meta-analysis of training studies. Psychol. Bull. **139**(2), 352–402 (2013)
16. Uttal, D.H., et al.: The malleability of spatial skills: a meta-analysis of training studies. Psychol. Bull. **139**(2), 352 (2013)
17. Wai, J., Lubinski, D., Benbow, C.P.: Spatial ability for STEM domains: aligning over 50 years of cumulative psychological knowledge solidifies its importance. J. Educ. Psychol. **101**(4), 817–835 (2009)
18. Yu, M., Ye, Z., Liu, Y., He, Y., Wang, C.C.L.: Lineup: computing chain-based physical transformation. ACM Trans. Graph. (TOG) **38**(1), 11:1–11:16 (2019)