AeroRigUI: Actuated TUIs for Spatial Interaction using Rigging Swarm Robots on Ceilings in Everyday Space

Lilith Yu*
University of Chicago
Chicago, Illinois, USA
lilithyu@uchicago.edu

Chenfeng Gao*
University of Chicago
Chicago, Illinois, USA
jessegao7@uchicago.edu

David Wu
University of Chicago
Chicago, Illinois, USA
davidwu26@uchicago.edu

Ken Nakagaki
University of Chicago
Chicago, Illinois, USA
knakagaki@uchicago.edu

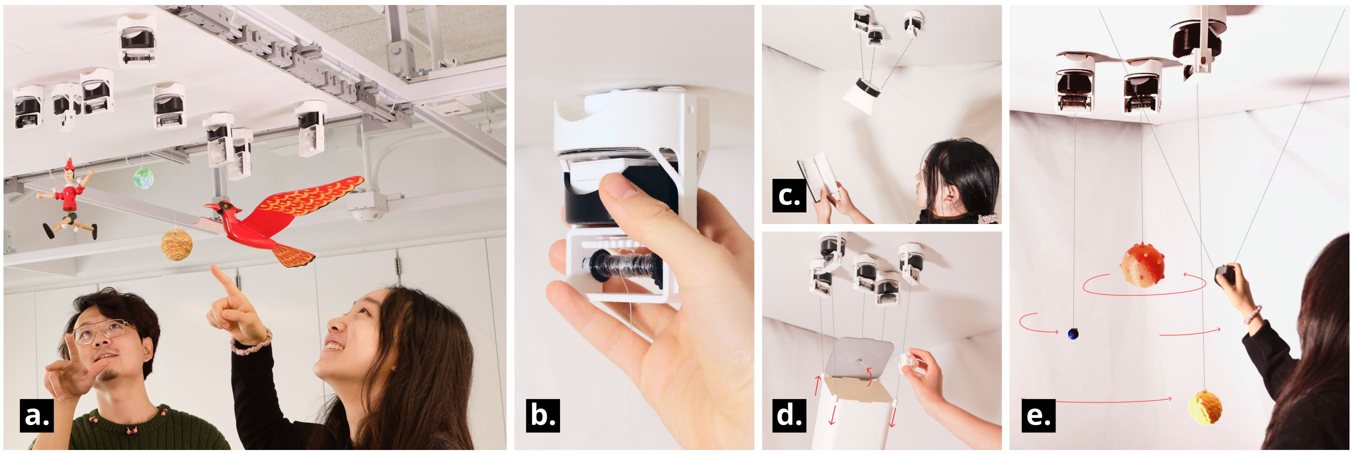


Figure 1: AeroRigUI: (a) users are tangibly interacting with everyday objects, controlled in the air by (b) *RigBots* – self-propelled swarm robots with a reeling mechanism. AeroRigUI provides *controllability* of objects and *deployability* in everyday spaces, allowing for applications such as (c) dynamic lighting, (d) tiltable storage container, and (e) interactive astral representation.

ABSTRACT

We present AeroRigUI, an actuated tangible UI for 3D spatial embodied interaction. Using strings controlled by self-propelled swarm robots with a reeling mechanism on ceiling surfaces, our approach enables *rigging* (control through strings) physical objects' position and orientation in the air. This can be applied to novel interactions in 3D space, including dynamic physical affordances, 3D information displays, and haptics. Utilizing the ceiling, an often underused room area, AeroRigUI can be applied for a range of applications such as room organization, data physicalization, and animated expressions. We demonstrate the applications based on our proof-of-concept prototype, which includes the hardware design of the rigging robots, named RigBots, and the software design for mid-air object control via interactive string manipulation. We also introduce technical evaluation and analysis of our approach prototype to address the hardware feasibility and safety. Overall, AeroRigUI enables a novel spatial and tangible UI system with great controllability and deployability.

CCS CONCEPTS

• **Human-centered computing** → **User interaction devices**.

KEYWORDS

Actuated Tangible User Interface, Spatial User Interface Display, Human Robot Interaction

ACM Reference Format:

Lilith Yu, Chenfeng Gao, David Wu, and Ken Nakagaki. 2023. AeroRigUI: Actuated TUIs for Spatial Interaction using Rigging Swarm Robots on Ceilings in Everyday Space. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3544548.3581437

1 INTRODUCTION

Aerial rigging is a well-developed, widely used stage technique for suspending and controlling performers in the air, specifically seen and performed in aerial circuses and dance performances [10] or in Broadway stage productions [33]. In these aerial arts, performers are rigged in mid-air using ropes and strings attached to the ceiling, and their 3D position can be controlled freely [67]. The aerial rigging technique can turn a conventional 2.5D stage into a 3D spatially immersive show, giving the audience the illusion of actors jumping high, levitating in the air, or flying to the moon. The technique allows for the actors to move dynamically in 3D space,

*Both authors contributed equally to this research.

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3581437

swinging from left to right, front to back, and low to high, which gives intrinsically dynamic physical expressions. Additionally, the technique is often used to effectively attract and move audiences with its quality of aesthetic expression by defying gravity and performing beyond the floor, the conventional stage.

In our research, we consider that the quality and benefits of aerial rigging can be applied to HCI research, specifically in the domains of spatial user interfaces and tangible UIs. We believe that the technique has significant potential to enrich our physical space with dynamically moving 3D objects to provide users with tangible, physical, spatial, and dynamic experiences. It can even add extra aesthetic qualities to everyday life found in aerially rigged performances. We envision a future of our everyday environments similar to a theater environment with aerially rigged objects, dancing and floating in 3D space.

In this paper, we introduce AeroRigUI, a novel type of actuated tangible user interface (A-TUI), that leverages a dynamically moving rigging robotic system on a ceiling surface to control physical objects in mid-air. Specifically, we deploy mobile swarm robots onto overhanging ceiling surfaces, allowing them to actuate reeling strings and control objects in the air. Our system can control the 3D position (x, y, z) of rigged objects using a single string, as well as the orientation (yaw, pitch, roll) when using multiple strings.

While prior research in HCI has proposed spatial tangible displays and user interface systems with different engineering techniques such as magnetic levitation [37], acoustic levitation [43], drones [51] or string-based vertical object control [12], our approach has unique advantages in **controllability** (e.g. lifting heavier objects, 6 DoF control, large control volume), as well as **deployability** (e.g. less noise, relatively easy to install on ceilings), that open up novel interaction design and application spaces.

In our paper, we introduce the design space of AeroRigUI that overviews our general approach, and enabled interaction design opportunities which provide new affordances and haptic design. To explore the design space, we implement a proof-of-concept prototype based on two self-propelled robots equipped with a custom design reel mechanism, referred to as *RigBots*. Our implementation also incorporates software to manipulate rigged objects' positions, which are processed with our robot movement calculation pipeline. We have conducted a series of technical evaluations to model and understand the hardware capability, which is crucial to assess and improve the system's usability with regard to safety and accuracy. This study has provided us with improvement in the design and control of our prototype. With the proof-of-concept implementation, we demonstrate applications to present the capability of AeroRigUI's spatial interaction design opportunities. The applications include room organization, interactive data physicalization, and animated crafts / kinetic art. Overall, AeroRigUI contributes to the spatial user interface design by enriching our physical environment with computationally controlled A-TUI systems that feature its strength in *controllability* of mid-air objects for interaction, and *deployability* of the approach for everyday space.

Our contributions include:

- An introduction of AeroRigUI, a general approach for designing 3D spatial tangible interaction based on ceiling swarm robots rigging physical objects in mid-air.
- A concept and design space of AeroRigUI that lays out the overall design architecture, actuation capabilities, and interaction design.
- A proof-of-concept implementation based on commercially available wheeled robots, and software development with a user control workflow and robot movement calculation pipeline.
- A model and evaluation of AeroRigUI and RigBot prototypes to reveal the hardware feasibility and guide for improving usability and safety.
- Applications to demonstrate the interactions design capability and use cases of AeroRigUI.
- A discussion of the limitations and future work to share future research opportunities.

2 RELATED WORK

AeroRigUI's contributions build upon prior research in (1) Actuated and Mid-Air TUIs, (2) String-driven UI and Kinetic Displays, (3) Swarm UIs, and (4) Ceiling Robots.

2.1 Actuated and Mid-Air TUIs

Advanced research and development in actuated and shape-changing tangible UIs has been explored in the past few decades, where researchers sought to embody dynamic digital computation with mechanically actuated objects to develop novel physical interaction [2, 11, 47, 48, 50]. Within such efforts, developing haptic and tangible interfaces in mid-air has been explored to bring the physical and tangible interaction not only onto the desktop but also into the 3D space. While many *graphical* systems have been developed to create 3D volumetric graphical images in mid-air, researchers have worked to *physically* render digital information in the 3D space. To add a sense of touch to the graphical image in the air, haptic researchers have developed non-contact haptic systems to provide haptic feedback via acoustic and aerial actuators [9, 55, 58]. While such non-contact haptics approaches lack tangibility, other researchers have investigated developing interfaces that manipulate physical objects mid-air. These interfaces utilize levitation technologies such as acoustic ultrasonic levitation [19, 20, 43, 44], air jet [3, 32, 58, 72], magnetism [37], or drones [6, 51]. Here, the levitating objects bring about mid-air 3D displays that allow for 3D and spatial tangible displays and interactions separate from that of tabletop TUIs [60].

AeroRigUI draws inspiration from these 3D mid-air A-TUI systems but introduces a novel approach to this research stream with *aerial rigging*: a technique to suspend objects in mid-air controlled by reeling strings using swarm robots on the ceilings. Our system creates new tangible and haptic interactions that cannot be done with previous levitation techniques. Through mobile robots and ceiling support structure, compared to other approaches, AeroRigUI features **controllability**, and **deployability**, which are detailed at the end of this section.

Additionally, other works have been investigated to enable room-scale actuation, mainly for Virtual Reality. These approaches employ technical means such as floor-robots [62], ceiling-fixed moving belts [5], or floor-embedded actuating pins, [24, 64, 66]. Our work partially contributes to this stream of room-scale haptics through a novel approach that employs swarm mobile robots on ceilings.

2.2 String-based UI/Kinetic Display

AeroRigUI's approach is informed by string-based kinetic sculptures such as *Kinetic Sculpture - the Shapes of Things to Come* by ART+COM [4], *Morphing Cube* [69], and *BLOOM SKIN* [68]. By controlling the position of suspended objects in mid-air, these kinetic sculptures provide graceful string-based actuation from above, display ambient information through actuated rigged elements, and allow users to walk amidst the 3D display, becoming fully immersed in it.

In the domain of HCI, string-based actuation is commonly used in the domain of haptics to render virtual forces [7, 14, 52]. Among them, the SPIDAR haptic user interface has a history of development based on a general approach of controlling motorized string with a reeling mechanism mounted on a frame to provide multi-directional variable force feedback to users' fingertips or hands [53]. While earlier developments were designed to be mounted on users' body parts in desktop areas, some variations of SPIDAR were introduced to control graspable objects with 6DoF force feedback [23], or to scale up to a room-scale force feedback interface [7]. We were inspired by the string-based actuation of SPIDAR that provides granular multi-directional force feedback, and intended to apply this actuation technique for spatial A-TUI systems.

In addition to haptics, most recently, *STRAIDE* [12] was presented as a string-based actuation approach to 2.5D shape-changing interfaces that allow for interaction with attached elements that can be actuated vertically (up-and-down). *STRAIDE* is designed with self-contained actuation towers that can be rearranged into different forms, such as an array, line, circle, etc.

While many of the above approaches are based on reeling mechanisms fixed to certain locations, our approach in AeroRigUI introduces the employment of swarm robots to flexibly and dynamically control the position of the reeling mechanism itself to provide richer controllability of the rigged/suspended objects. As prior research has largely investigated objects moving only up and down (1D), our approach can control *x, y, and z* (3D), when controlled with a single string, and can be extended to control *yaw, pitch, and roll* for 6D controllability, using multiple strings.

2.3 Swarm User Interface

In recent years, a variety of research has been explored on the topic of Swarm User Interface (SUI) [35], which employs a swarm of self-propelled robots to communicate digital information tangibly. Researchers have proposed different applications with SUIs to provide force feedback [30, 65], to bring modularity with passive mechanical shell attachments [40], to assemble modular bricks [73], or to control on stages for appearing and disappearing expressions [41]. However, many of these systems have been constrained to table-top surfaces as an interaction area, with the exception of UbiSwarm [29], which partially explored on-wall surfaces. By deploying swarm mobile robots to the ceiling surfaces, we expand a whole new dimension as an interaction area in the room-scale 3D space.

2.4 Ceiling Crane, Robots and Puppeteering Robots

Prior research and technology outside of HCI have also extensively informed the design and functionality of AeroRigUI. For example, ceiling mount cranes (or overhead cranes) are machines that are integrated into industrial environments to help workers to move large, heavy objects for manufacturing or assembling [36, 45][1]. These cranes commonly consist of environmentally installed rails and, actuating reeling units, hoists, that move overhead across the space to help manipulate objects.

Robotics researchers have explored the development of robots that can locomote on unconventional surfaces such as walls [22, 31, 70] and ceilings [8, 17, 26, 59]. They have explored different techniques to stick to the surfaces, such as dry adhesives [31], vacuuming [71], or granular jamming [16]. Magnetic adhesion – employing magnets to stick to ferromagnetic surfaces [22, 57, 70] – is a common approach in robotics that AeroRigUI also employs for wheeled robots. The idea of developing ceiling robots that manipulate our physical environment is presented by robotics researchers [17, 18, 26, 54], although these systems tend to be bulky and cumbersome to be easily deployed to physical environments. The closest robotic implementation to our system are works by Jochum et. al., and Sato et. al.[26, 54], which requires additional locomotive hardware above ceiling surfaces to hold the robots. Our implementation is relatively easier as it leverages ferromagnetic surfaces and magnet-embedded mobile robots.

Additionally, the robotics community has employed string-based actuation to manipulate hanging marionettes using robotic arms [27, 74]. Research by Jochum et. al., to use of on-ceiling wheeled robots with reeling strings is the most relevant robotic design to ours, where they have successfully modeled, and controlled marionette limbs with ceiling robots [26]. Our work, in addition to developing a relatively compact and accessible design, explores an actuation system for user interface design by comprehensively defining and demonstrating interaction design opportunities and application spaces.

2.5 Positioning of AeroRigUI and Challenges

Here, we highlight the key characteristics of AeroRigUI as a spatial A-TUI system, categorized in *controllability* for display and interaction as well as *deployability* to everyday space.

Controllability features the capacity and effectiveness of mid-air object control. For example, our approach can control objects in 6DoF (x, y, z, row, pitch, yaw), and relatively heavier objects (approx. 1 kg) in mid-air. It can also control objects in a room-scaled space (both in horizontal and vertical ranges). On the other side, compared to other works, there are weaknesses in the resolutions of the displayed objects and the maximum number of objects that can be controlled. **Deployability** encompasses the potential to be deployed to the physical environment. For example, with the ferromagnetic surface and mobile robots, it can be easily embedded in the physical environment with efficient use of space.

There are some critical technical challenges for the approach in AeroRigUI, namely, 'safety' for robots to drop with excessive

[1]https://en.wikipedia.org/wiki/Overhead_crane

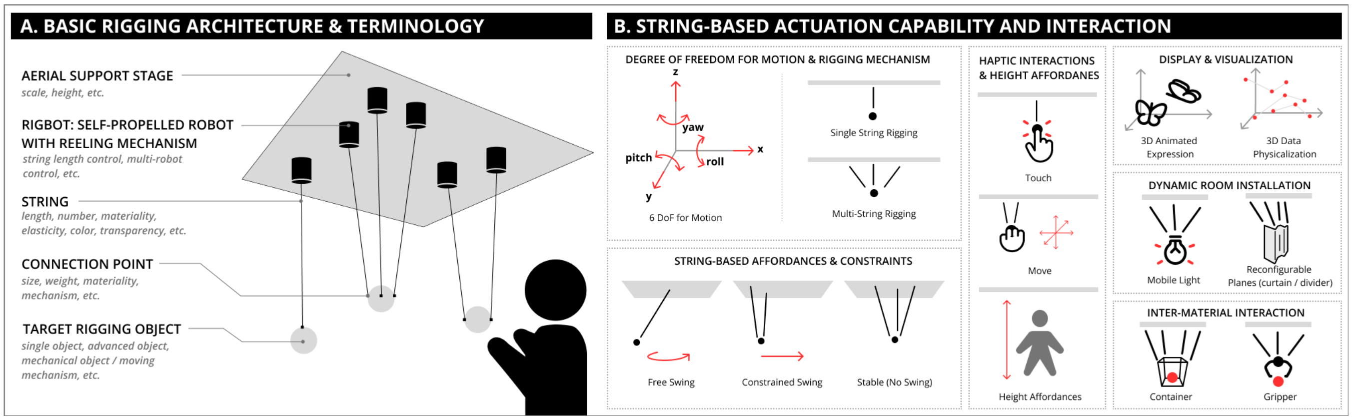


Figure 2: Design Space of AeroRigUI.

force, or 'swing' to reduce the accuracy of control, which are later addressed in our paper.

Overall, the general approach of AeroRigUI, using self-propelled reeling robots on the ceiling surface, should open up novel features in controllability and deployability that contributes to the state-of-art in 3D spatial haptic and tangible UI design.

3 AERORIGUI

We first introduce the design space of AeroRigUI (Figure 2). This is intended to share the general architecture design as well as interaction design characteristics for fellow researchers and developers.

3.1 Basic Rigging Architecture

As in Figure 2(A), AeroRigUI consists of the (1) aerial support stage, (2) self-propelled robots equipped with reeling mechanism (referred to as *RigBot* in our paper), (3) strings, (4) connection points, and (5) target rigging object.

Aerial Support Stage. The aerial support stage defines the space that the swarm robots can move on. The scale and height of this structure can be adjusted to create small or room-scale interactions.

RigBots. *RigBots* are self-propelled swarm robot modules that move along the aerial support stage and control the strings with their included reeling mechanism. Multiple RigBots can be controlled simultaneously, and they can move and rotate to control the string's location and length.

Strings. The strings are used to keep objects in the air. String lengths can be dynamically adjusted by the RigBots. The number of strings connected to each object can be adjusted to create different swinging or stable motions. While, in our prototype, we mainly use transparent fishing strings to make strings less visible from a user interaction perspective, the string's materiality, elasticity, color, and transparency could be further explored for customized haptic and visual expressions.

Target Rigged Object. Target rigged objects are the main tangible body for user interaction, controlled by the RigBots via strings. Changing the number and content of the target rigging object may bring about different interactions, such as display and visualization, inter-material interactions, and haptic interactions.

Connection Points. Connection points are 3D points where the string is attached to the object. In the system, one object can be attached to one or multiple RigBots via connection points to perform different movements.

3.2 String-based Actuation Capability and Interaction

A range of actuation capabilities and interactions can be covered with the generic configuration of AeroRigUI (Figure 2(B)).

Degrees of Freedom for Motion and Rigging Mechanism AeroRigUI can translate objects in X and Y directions by RigBots moving on the aerial support stage, and also move in the Z direction by adjusting the string length. When the objects are controlled by more than three strings, they can additionally control more degrees of freedom, including yaw, pitch, and roll[34].

String-based Affordances and Constraints One of the unique design elements, we explore with AeroRigUI, is how we can control the objects' affordance and constraints by customizing the number of controlling strings. With single-string rigging, the objects can *swing freely*. This affordance may be useful for dynamic interactions with fewer constraints that allow users to move freely within the single-string constraints. Secondly, when objects are connected with two strings, the rigged object would be *constrained for 1D linear swing*. This also allows for haptic constraints that, for example, are used for 1D slider input with tactile constraints. Thirdly, when objects are constrained with more than three strings, their motion is most constrained to be stabled in a 3D position.

While there is an implementation trade-off that making objects stable in the air without swing requires more RigBots and strings (which would take up more space on the ceiling), designers of AeroRigUI could utilize and customize the swinging constraints according to their intended interaction applications and balance this trade-off.

Haptic Interactions and Height Affordances The rigged objects in the air can afford different types of haptic interactions such as *touch* and *move* as the objects become tangible and graspable UI elements. With the above-mentioned different string constraints, the affordance for users to touch and move the objects can be customized.

Additionally, with the capability of controlling the object's position anywhere in the 3D space, their positioning can afford advanced bodily interaction, especially with different heights, which we call *height affordance*. For example, positioning objects high near the ceiling can infer that they should not be touched by users, and bringing in lower positions could induce different actions for users (e.g. chest height for touching, leg height for kicking). Such interaction techniques shall be practical for interaction design, such as kinesthetic learning applications.

Display / Visualization As for the interactive utility of AeroRigUI, it can be used for display and visualization. It can support different kinds of visual and tangible 3D expressions such as *3D Animated Expression* to convey semantic physical and dynamic character, as well as *3D Data Physicalization* to physically represent abstract data [25].

Dynamic Room Installations With AeroRigUI, everyday room installations can be dynamically reconfigured. For example, *lighting* on a ceiling can be controlled to adapt to user interactions, and *reconfigurable plane* can be controlled to act as dynamic space separators, curtains, or projection screens.

Inter-material Interaction Another utility of AeroRigUI is Inter-material Interaction, which is a mode of interaction for actuated and shape-changing UIs to control other passive objects [15]. With AeroRigUI, special types of objects can be rigged to act on other objects. For example, AeroRigUI can control *containers*, moving them towards the users to pick up objects and store them; AeroRigUI may also actuate *grippers* to actively handle, grab and release passive objects in a room.

4 HARDWARE IMPLEMENTATION: RIGBOT PROTOTYPE DEVICES

In this section, we introduce the implementation and technical details of the proof-of-concept prototype of AeroRigUI. The overall system of our prototype is based on RigBots, an overhanging ferromagnetic ceiling stage, and a computer to control the system (Figure 3).

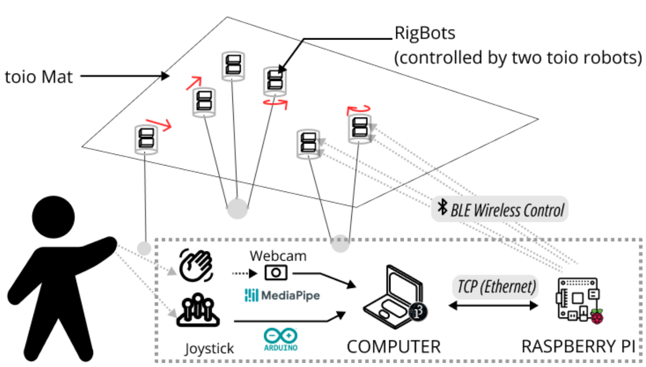


Figure 3: System Overview of Implementation of AeroRigUI

4.1 RigBot Hardware Implementation

We designed two RigBot modules, (1) the regular RigBot, and (2) a heavy-load edition of RigBot. The heavy-load edition has a higher rigging weight but a slower reeling speed. In this section, we introduce the implementation of the RigBot hardware.

Figure 4(A) and (B) show the design of the regular RigBot, a cylinder shape composed of two embedded toio robots and 3D printed mechanical parts. It is 64mm diameter and 82mm tall. One toio robot handles locomotion on the ceiling surfaces, while the other is used to control a reeling mechanism by spinning on top of the other toio robot. RigBots are equipped with magnets so it can be easily sticked to ferromagnetic ceiling surfaces. To increase the payload capacity of the RigBot, we intended to maximize the magnetic adhesive force between the RigBot and the ceiling. We attached two disc-shaped neodymium magnets (6mm diameter, 1.5mm thickness, 0.39kg pull force with 0.5mm distance from the ceiling) on the bottom of each toio robot (as implemented in HERMITS[40]), and embedded another two more cylindrical neodymium magnets (9.5mm diameter, 12.5mm thickness) inside the 3D printed shell as depicted in Figure 4(A). These additional magnets are positioned 2.5mm from the ceiling (we found this gap to be the shortest distance away from the ceiling while ensuring the magnets does not hinder RigBot's movement). The original pull force of this additional magnet is 4.77kg. However, with the distance of 2.5mm from the ceiling, its pull force is now 0.77kg

Figure 4(A) shows an exploded-view drawing of a RigBot, that details the internal mechanism. For the reeling toio, the spinning motion was translated to the reel via bevel gear mechanism. The reel is threaded with 4+ meters of clear nylon fishing wire (0.3 mm diameter, 7.6 kg load capacity). Due to the toio's capability in localizing their positions and orientation, the robots are able to identify their position on the ceiling and the amount of reeling rotation. The enclosure cover was designed to hide the reeling mechanism, though, in our work, we demonstrate without the cover to present the mechanism.

Additionally, we also designed and implemented a heavy-load RigBot module capable of controlling heavier objects. The heavy-load edition has the exact same design as the regular RigBot except for a changed gear ratio (Figure 4(C)). We leveraged gear reduction to create a higher output torque for the robot to rig heavier objects. While the regular edition has a 2:1 gear ratio, the heavy load edition has a 1:5 gear ratio. However, the lower gear ratio comes with the trade-off of a reduced rigging speed–it will take much longer time for the object to achieve its target Z position. Another trade-off of the heavy load edition is its lower payload capacity due to its heavier weight (details outlined below). Users could utilize different RigBot editions according to the application and rigging objects.

As for the specs of our RigBot prototypes, we have measured three different weights for each of the RigBot editions (regular and heavy-load). The regular RigBot weighs 0.151kg. Its payload weight, evaluated by steadily pulling it down with a force gauge, is 0.940kg (9.21N). The rigging capacity, the weight the rigging mechanism can pull up, was measured as 0.260kg (2.55N). For the heavy RigBot module, the RigBot itself weighs 0.163 kg, its payload capacity weight is 0.860kg (8.43N), and it has a rigging capacity of 0.718kg (7.04N) (2.76x the rigging capacity of a regular RigBot). In section 6, we investigate, analyze, and evaluate the payload capacity in different conditions.

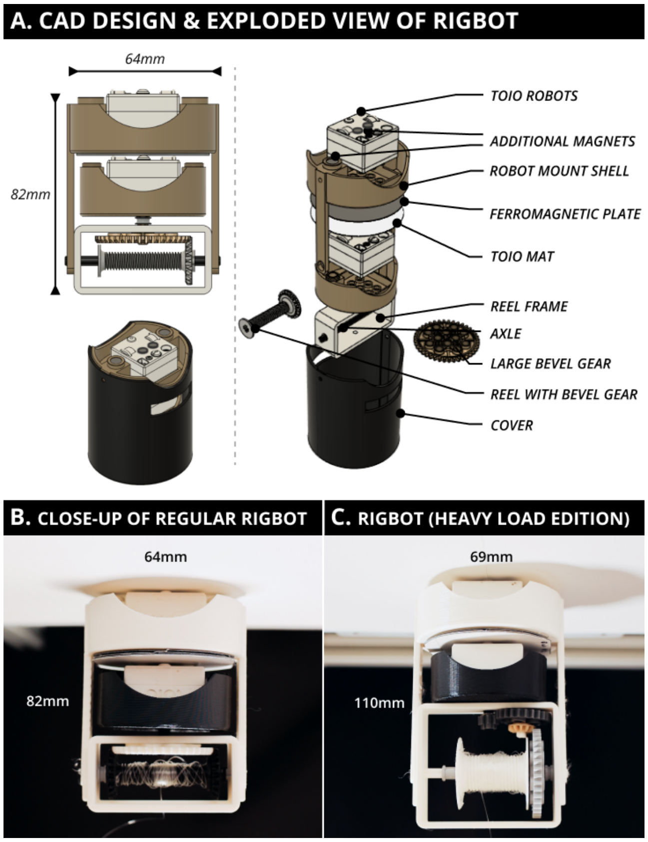


Figure 4: Design of the RigBot module. A) CAD Design and exploded view of its parts. B) Close-up view of RigBot regular edition on aerial support stage. C) Close-up view of RigBot (heavy load edition).

4.2 Aerial Support Stage and Overhanging Magnetic Toio Mats

We designed an aerial support stage for the RigBots to travel on, as seen in Figure 5. The magnet mounts hold modular toio mats with ferromagnetic sheets [13] on the ceiling using magnetic adhesion. We put 12 pieces of A3-sized toio mat together, creating a 1260mm x 1188mm aerial stage. The modular construction provides flexibility and deployability in everyday space.

4.3 Rigged Objects

Different rigging objects were fabricated and prepared for our prototype. We outline them in three categories: (1) single-string rigged objects, (2) multi-string rigged objects, and (3) mechanical rigged objects. These objects range from 1 gram to 318 grams as annotated in Figure 6.

Single-string Rigged Object (for 3DoF) - Simple objects, such as planets in the solar system and office supplies (Figure 6a), are rigged with a single string. These rigged objects can move with 3 degrees of freedom (x,y,z) controlled by a single RigBot.

Multi-string Rigged Object (for 6DoF) - We use multiple strings (2-5) to rig objects such as curtains and lights (Figure 6b). With multiple strings, we are able to control not only the object's position but also its orientation (yaw, pitch, roll).

Mechanical Rigged Objects - The mechanical rigged objects include a gripper, a trash container, a marionette, and a puppet bird (Figure 6c). On top of controlling these objects in 6DoF, we use the strings to actuate specific parts of these objects to produce advanced tasks, interactions, and expressions.

The functionalities of each rigged object are further demonstrated in our applications.

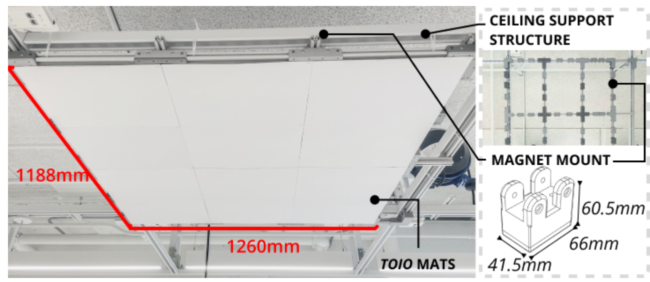


Figure 5: Design of Aerial Support Stage: Overview of Stage, Ceiling Support Structure and CAD model of Magnet Mount

4.4 Modular string connectors

We designed modular string connectors to (1) increase usability for the users, making objects easily attachable and detachable, and (2) ensure the safety of our system. Two types of connector mechanisms were developed: (1) magnet and (2) hook. Users may choose between the magnet or the hook depending on the rigged object, as the magnet suits high-swinging applications, while the hook suits better for lifting heavy objects.

To ensure the safety of our system, for the magnetic connectors, we design a magnetic connector whose magnetic adhesive force is under the payload capacity of the RigBot. Currently, a RigBot's payload capacity is 0.94 kg. For our magnetic connector, we use a cylinder-shaped magnet of 6.35mm diameter and 6.35mm thickness. We covered the magnet with a 3D-printed enclosure with a 1.2mm thickness. With the added thickness, its magnetic adhesion force is decreased from 1.88 to 0.8 kg. This is right under the RigBot's payload capacity. Thus, if a user pulls on the rigged object with a force that exceeds our system's payload capacity, this mechanism will ensure that the object becomes detached from the magnetic connector and the rest of the AeroRigUI system. This prevents a RigBot from falling off the ceiling, which can be a safety hazard.

5 SOFTWARE TO CONTROL ROBOTS

We have developed software including a GUI that allows users to control rigged objects and a control pipeline that calculates robot

Figure 6: (a) examples of *single-string rigged objects*: solar-system planets, pen holder; (b) examples of *multi-string rigged objects*: reconfigurable curtain, rigged TUI, moveable light; (c) examples of *mechanical rigged objects*: trash container, puppet bird, gripper, marionette. (d) types of modular string connectors: hook, magnet, and magnet with hook

movement based on the user inputs. The software is developed using Processing (Figure 7).

5.1 User Workflow

To flexibly control different rigging objects and conditions, we design a user workflow that consists of two phases, (1) setup & prep and (2) control (Figure 7A). While phase (1) allows users to set up the physical hardware as well as the config parameters of rigging objects for the software, phase (2) allows for real-time control based on the parameters set in phase (1).

In the **setup & prep phase**, the user first inputs the number of target rigged objects in the system, the number of connection points on each object, and the distances between each connection point for each object. Then, the user may command the robots to calibrate the string length by reeling up their strings to the top, before attaching the object for control.

In our GUI, we have also developed **Payload Tool** for users to verify the payload conditions on the left side of the GUI before they start to rig objects, to help users to improve safety, minimizing the risk of the RigBots to drop from the ceiling. This tool was derived from our models of payload capacity created based on our technical evaluation (detailed in the following section 6). In this tool, as the user inputs the rigging object weight and the number of RigBots, our software calculates the payload capacity of the system and verifies that the object weight is safe enough to be rigged. The software also derives the maximum acceleration and string angle of the system based on the payload capacity. The GUI also informs the user whether the given condition is safe with the following criteria: if the object is under the payload capacity with a maximum angle less than 15 degrees, we consider the system to be safe; if the object is under the payload capacity with a maximum angle greater than 15 degrees, we give users a caution message to remind them that they may run into safety risks (we chose 15 degrees as rigged objects generally don't swing more than 15 degrees, with the exception of special cases when users purposefully pull the rigged object at a large angle); and if the object exceeds the payload capacity, we give the user a warning and set the maximum acceleration to 0 so that RigBots will not be able to move. (*Refer to section 6 about how angle and acceleration affect the payload capacities.)

In the **control phase**, we have implemented a GUI system that shows 3D visualization for users to simulate the relationship between rigged objects and RigBots, and a control panel for users to control the state of objects, depicted in Figure 7A. The right side of the control GUI features basic object control parameters. Users may switch between objects to control, choose between real-robot control and simulation on visualization, etc. With this GUI control, users can control the rigged object's position without considering each robot's position, thanks to our back-end pipeline for calculating robot movement, which is detailed in the next.

Users may also choose between the three modes of RigBot movement and adjust the RigBot maximum acceleration. The range of the acceleration control bar is capped at the derived maximum acceleration from the payload capacity.

5.2 Robot Movement Calculation Pipeline

We create a calculation pipeline in Processing that maps users' inputs into concrete robot movements (Figure 7 B). Overall, we abstracted our pipeline into three stages: (1) Target and Config, (2) Abstracted Kinematics, and (3) Concrete Robot Control. In the **Target and Config** stage, we take the user inputs from the setup & prep phase of the user workflow as static variables, and the inputs from the control phase as dynamic variables. From the static variables, we instantiate abstract coordinates called *connection points* to mark the points of contact between the object and the string. And from the dynamic variables, we instantiate the target object conditions (X, Y, Z, yaw, pitch, roll, and stability) – instructed by the GUI control. Then, at **abstracted kinematics** stage, these inputs are passed throw the target fitting process, where new connection points are calculated based on the target object position. The next stage maps the X and Y values of the new connection points to *RigBot positions*, and the Z values to *string length*. Lastly, during the **concrete robot control** stage, the new RigBot positions and string lengths are passed in to generate translation and rotation commands for each toio robot, which are wirelessly communicated via Bluetooth with a Raspberry Pi controller [40].

5.3 Additional Control Methods: Hand-tracking and Joystick

Besides the control with the GUI, we implemented two supplemental input methods to control and interact with AeroRigUI: (1) hand

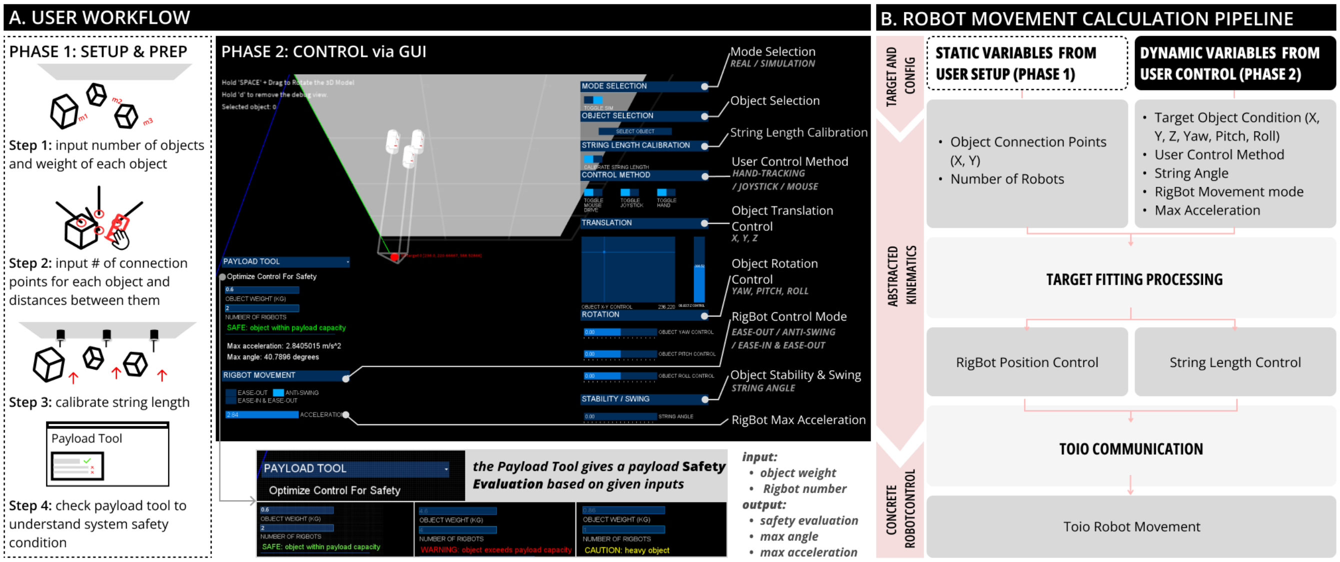


Figure 7: User Workflow and Robot Movement Calculation Pipeline.

tracking and (2) joystick control. On the GUI, these optional control modes can be activated. These control methods were employed to demonstrate some of our applications.

Hand-tracking: For hand-tracking, we use Google's MediaPipe[2], which can track the 3D positions of the hand joints (21 points in total) using an RGB image input. The tracking software, developed in Python3, communicates the tracked coordinate points to our Processing software via a local socket. We mounted a USB camera on the ceiling or the floor, depending on the area in which we track the hand. Adapting to different mounting configurations, we manually calibrate and convert the input finger positions to be mapped on our 3D visualization, as shown in Figure 8.

With this setup, we were able to develop different methods to control objects with different hand gestures, such as mapping the 6DoF of the palm to the orientation of the toy bird (Figure 8a), or mapping finger positions to the limbs of a marionette (Figure 17c). We also implemented detecting tangible interaction with rigged objects by developing collision detection between the hand and the rigged object. In this way, our system could detect how users were moving a graspable object in the air (Figure 8b).

While future implementations could employ an advanced tracking system with highly accurate 3D hand-tracking and a wider tracking area (e.g. OptiTrack [3]), our implementation is sufficient for a preliminary exploration and demonstration of a range of applications.

Joystick Control: In addition to hand-tracking, we also employed a joystick controller, embedded with three joysticks. Their inputs were detected by an Arduino UNO and passed to Processing via serial communication. As each joystick has 2DoF reading capability, resulting in a total of 6DoF inputs, we used three joysticks to control one target object's 6DoF (x, y, z, roll, pitch, yaw). When there were multiple objects, a GUI was used to switch between the objects to be controlled. Each joystick was embedded with an additional press switch, used to activate additional mechanisms, for example, the grab & release mechanism of the gripper (Figure 6b), and the open & close mechanism of the trash can (Figure 6c).

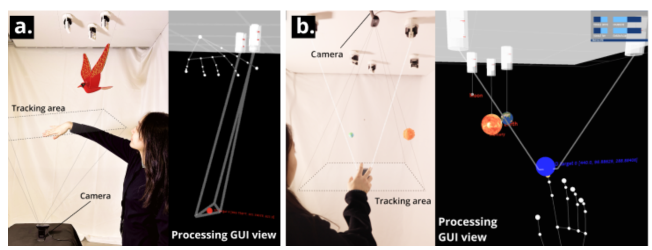


Figure 8: (a) hand-tracking control for gesturally manipulating a bird toy, (b) detecting touch and moving tangible objects in the air via hand-tracking.

6 MODELING AND EVALUATING AERORIGUI AND RIGBOT PROTOTYPE

We found several crucial technical factors that need to be modeled, analyzed, and evaluated to better understand the capability and limitations of our approach toward a usable interactive system. Specifically, we assessed and analyzed *(1) payload capacity, (2) swing, and (3) DoF control accuracy* to ensure the safety, stability, and usability of our system as a physical display and interaction system.

Such an assessment will help determine and optimize the software control and hardware setup (e.g. to help users decide the weight of the objects to be rigged). Overall, we hope to provide generalized models and design/control guides for future researchers and users. In the section below, technical factors (1)-(3) are studied and evaluated to understand the system's capacity through our prototypes.

[2]https://google.github.io/mediapipe/solutions/solutions.html
[3]https://optitrack.com/

6.1 Payload Capacity

6.1.1 Definition and Basic Model of Payload Capacity. **Payload** is one of the most important technical factors we have identified to make AeroRigUI a robust and safe interactive system. We define **Payload** as *additional force the RigBot has to handle*, including the weight of rigged objects as well as external forces acting on the objects (e.g. force exerted from user pulling). Due to the nature of our approach – sticking mobile robots embedded with magnets onto ferromagnetic ceiling surfaces – there is a clear limitation in the payload the robots can sustain until they detach and drop from the ceiling, a huge safety risk. We define this as **Payload Capacity**: *the maximum force a RigBot can sustain before the RigBot detaches from the ceiling*. In this section, we intend to analyze how the payload capacity is affected by different technical factors based on the system's variables, so that we can provide insights into minimizing the risk of RigBots dropping from the ceiling.

Before discussing the extended technical factors, we first introduce the basic model for the payload capacity(Figure 9 left). As the illustrated model [a] shows, a RigBot sustains a payload P – consisting of the weight of a rigged object $m_o g$ and an external force $F_e xt$ (model [a1]). Generally, the Payload has to be smaller than Payload Capacity P_{cap} (model [a2]), to ensure the robots don't fall from the ceiling. Here, we define **Basic Payload Capacity**, P_{cap_b}, as a primitive payload capacity used to model further factors in payload capacity. $P_{cap_b}$is represented as $F_b - m_r g$, where the magnetic adhesive force between a RigBot and the ceiling is represented as F_b, and the gravitational force acting on the RigBot is $m_r g$ (model [a2]). Hence, larger F_b and smaller $m_r g$ results in larger basic payload capacity.

As reported in section 3.2, the basic payload capacity of our prototype was measured to be 0.94kg, or 9.4N. While our basic model shows the essential factors of payload capacity, in reality, we observed that a RigBot's payload capacity is also affected by other technical factors, including *[b] string angle*, *[c] RigBot acceleration*, and *[d] number of RigBots*. With this, we analyze and present a general model for each factor ([b-d]) below that is derived based on technical evaluations with our hardware prototype. Our findings are depicted in Figure 9 as well.

6.1.2 Payload and String Angle. In the AeroRigUI setup, the angle of the string, θ, can be varied in different cases(Figure 9 [b]). With our hardware prototype, we found that pulling a RigBot at an angle lowers the payload capacity due to introducing a horizontal component of the pull force to the RigBot, resulting in the robot to tip. To better understand how the angle affects the payload capacity, we have quantitatively evaluated it with our prototype hardware and developed a model.

Technical Evaluation:

To measure the relationship between string angle and payload capacity, we attached a Newton Force Meter Spring Scale (capacity 0-10N, readability 0.2N, accuracy +/- 1%) to the end of the string of a RigBot [b1]. We measured angles between 0-25 degrees with 5-degree increments. For each angle, we measured the average amount of pull force it took to break the magnetic attraction of the RigBot, whose results are reported in [b2]..

Model, Analysis, and Implication: With this result, we can derive a model [b3] whereas the string angle θ increases, the payload capacity drops with respect to a constant variable c_θ (in our case, this is -0.025 Newtons/degree).

This analysis of the string angle's relationship with payload capacity informs us to minimize the string angle to minimize the risk of RigBots dropping. A concrete solution is to develop kinematic constraints to limit the user from pulling the rigged objects at greater angles. Another solution is to reduce the swinging amount generated by robot motion control when the rigged object's weight is right under the payload capacity.

6.1.3 Payload and RigBot Acceleration. Since AeroRigUI is a dynamic system, where RigBots dynamically travel across the ceiling surface horizontally, it is important to consider how their motion affects the payload capacity. Through our preliminary observation, we found that increasing the acceleration a (depicted in [c]), has resulted in lowering the payload capacity. We believe that when the RigBot accelerates, the pull force from the rigged object increases due to the load moving in a non-inertial or accelerating frame.

Technical Evaluation:

To quantify how RigBot acceleration affects the payload capacity, we set up a RigBot with a fixed string length and attached weights (accuracy +/- 20g) [c2]. We moved the RigBot between two fixed points 25.0 cm apart, varied the RigBot acceleration, and observed at what weight the RigBot detached from the ceiling to measure the payload capacity. With five different accelerations in the range of toio's control capability, we made three measurements to calculate the average payload capacity for each acceleration. The result is depicted in [c2].

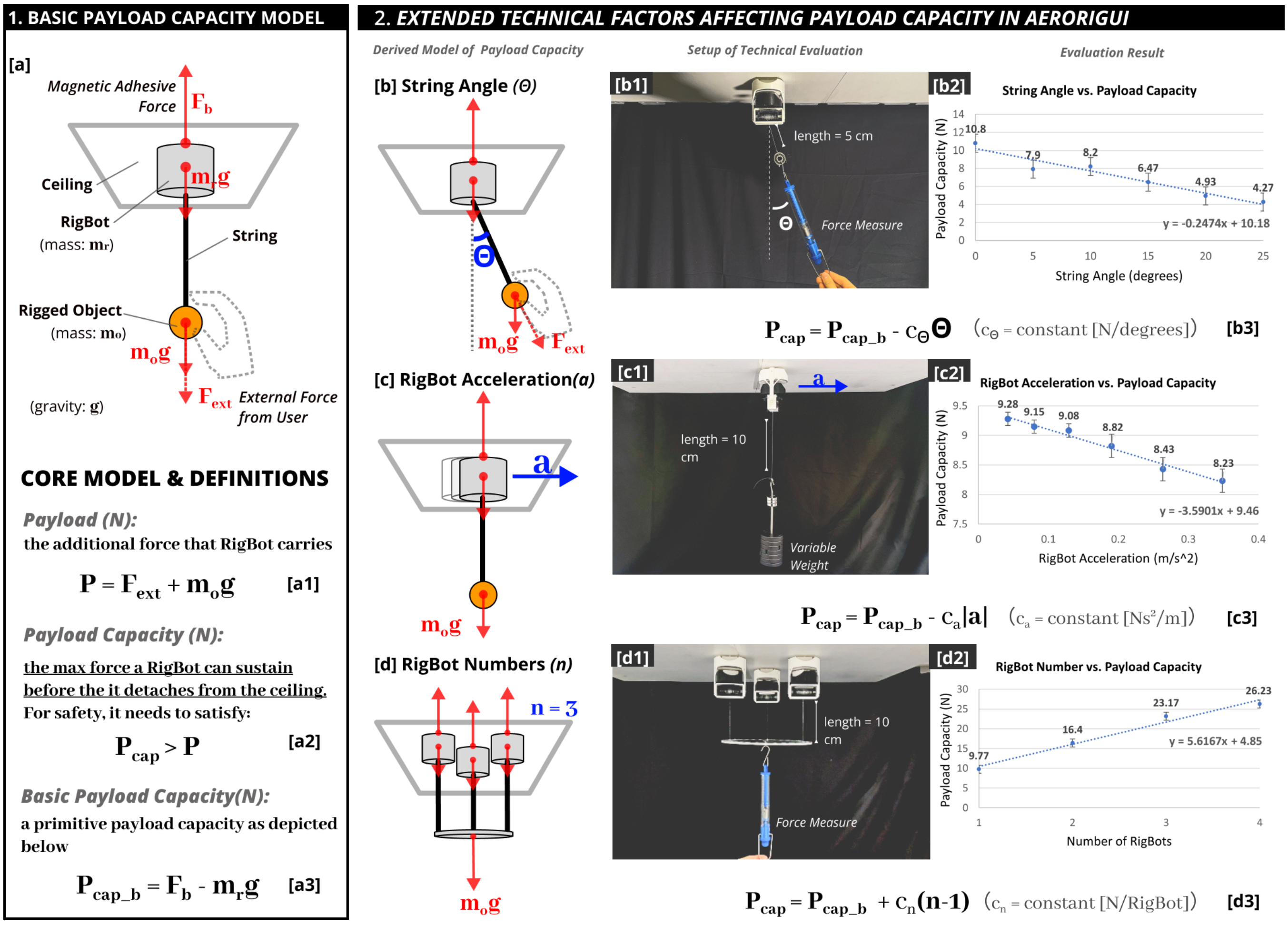
Model, Analysis, and Implication:

This result derives another model for the relationship between the RigBot acceleration and payload capacity [c3], where acceleration a negatively affects the payload capacity based on the constant variable c_a – measured as 3.59 Ns^2/m. This informs us that the system could minimize RigBot acceleration to avoid decreasing the payload capacity. This model can be further incorporated into our control software to limit the maximum acceleration variably depending on the weight of the rigged objects.

6.1.4 Payload and RigBot Number. One unique feature of AeroRigUI is that it allows an object to be rigged by multiple RigBots. This allows our system to exceed the basic payload capacity of a single RigBot and carry heavier objects, as depicted in [d]. To analyze this, we conducted a technical evaluation and derived a model.

Technical Evaluation:

We prepared a disc-shaped acrylic plate (diameter 20cm, weight 146g) that can be attached with 1-4 strings equidistant from each other. We varied the number of RigBots to rig this plate and attached a Newton Force Meter Spring Scale (capacity 10N and 20N; readability 0.2N; accuracy +/- 1%) at the center of the acrylic plate to measure the payload capacity with different numbers of RigBots [d1]. The strings were kept perpendicular to the ceiling, so the test was not affected by the string angles (to eliminate mixed factors from [b]). We recorded the force that triggered one of the robots to detach from the ceiling and calculated an average from three trials for each RigBot number, n, between 1-4. The result is shown in [d2].



$$P = F_{ext} + m_o g \quad [a1]$$

$$P_{cap} > P \quad [a2]$$

$$P_{cap\_b} = F_b - m_r g \quad [a3]$$

$$P_{cap} = P_{cap\_b} - c_\Theta \Theta \quad [b3]$$

$$P_{cap} = P_{cap\_b} - c_a |a| \quad [c3]$$

$$P_{cap} = P_{cap\_b} + c_n (n-1) \quad [d3]$$

Figure 9: Payload Capacity Model and Evaluation Diagram that includes [a] the core model and definition of basic payload capacity, and provides analysis of how three technical factors affect the payload capacity ([b] String Angle, [c] RigBot Acceleration, and [d] RigBot Number).

Model, Analysis, and Implication:
The evaluation result derives the model on the relationship between the RigBot number and payload force, where increasing the number of RigBots n by one increases the payload capacity by a constant variable of c_n – in our setup, measured as 5.6 $N/RigBot$. This result informs that even if a rigging object is heavier than the payload capacity of a single RigBot, the number of robots can be increased to accommodate the extra weight. Further, when designers anticipate a relatively strong pulling force from users, a rigging object can be rigged by more robots accordingly. While more robots increase ceiling space usage, a future developer of AeroRigUI could handle this trade-off, guided by our model.

6.1.5 Summary for Payload Capacity. To summarize, we identified three technical factors affecting the payload capacity variably. We derived models based on technical evaluations with our prototype. We incorporated these factors into our GUI tool to help users handle the payload capacity – as the GUI alerts users when the rigging object weight is beyond the payload capacity, displays the maximum angle for specific rigging objects, and limits the maximum acceleration depending on the rigged weight.

Future studies could look closer into understanding how these factors can be integrated into one unified model, and explore other factors affecting the payload. Furthermore, if we can build an accurate simulation model of the hardware, such a tool could help optimize the hardware design in the future. In our paper, the study is an important first step to providing insights into payload capacities.

6.2 Swing affected by RigBots' Motion

Swing is another factor that has to be inspected for understanding the limitations of AeroRigUI. As the general mechanical design of AeroRigUI is prone to swinging – especially when rigging an object with a single string – it is important that our system is designed and operated with a swing in mind. This is crucial for representing a 3D position of rigged objects dynamically. We observed that the many different ways the robots travel across the ceiling plane affected the swing differently, which we evaluated and analyzed.

Prior research in modeling crane control kinematics in an open-loop control system has addressed the challenge of keeping rigged (or craned) objects with less swing while the rigging point moves in horizontal directions [21], which is similar to our setup. Based on prior research, we have incorporated a control method to reduce the swing by controlling the horizontal motion of the RigBots. We call this control method *Anti-Swing Control* and evaluated it to understand how it minimizes the swing compared with the other two control methods.

6.2.1 Robots Movement Control Methods. We compared three control methods of RigBots, as shown in Figure 10A, which depicts the velocity curve across time while the robot is moving to a designated target from an origin point. All control methods have a cohesive parameter named *maxMotorSpeed*, which defines the maximum speed for the motors of the toio robot during the locomotion. To simplify our study and analysis, we designed all the controls to make a straight linear motion from an origin to a target.

The first control is *[a] Ease-Out Control*, which enables the robots to move towards the target coordinate by starting with maximum motor speed, gradually slowing down as it approaches the destination (we used a function named *aimCubeSpeed()* within an open source toio control code[4]). Secondly, *[b] Ease-in & Ease-Out Control* directs the robots to gradually increase the speed in the beginning and slow down when it gets closer to the target (we have utilized a movement control command named *Motor control with acceleration specified*, within toio API command [5] – with acceleration parameter of 5).

Finally, we have developed *[c] Anti-Swing Control* based on [21], which changes the speed according to a model decided by the string length, overall travel distance, and *maxMotorSpeed* – to minimize the swing of rigged objects. While this method is further described in [21] and our developed code is shared in a GitHub repo [6], we provide a brief explanation here.

As the velocity curve in Figure 10A[c] depicts (this curve was referred from [21]'s Figure 3b and 4b), during the acceleration phase, the RigBot undergoes a quick jerk period, in the beginning, to let the rigged object start moving within an acceptable swing angle. Then a constant acceleration motion will lead the RigBot to approach the maximum speed, defined as *maxMotorSpeed*. At the end of the acceleration phase, a damping period, opposite to the previous jerk action, is added to eliminate the swing angle that was made before. As for the deceleration phase, the RigBot's velocity is decreased linearly as the robot travels toward the target destination.

6.2.2 Technical Evaluation for Swing and Control Methods. We have measured the amount of swing for each control with three different *maxMotorSpeed* parameters, 45, 80, 115 – which are parameters defined in motor control command of toio [7]. The string length was varied between 30, 60, 90cm, and the travel distance was set at 60cm. The weight of the object was fixed to 21 g. We ran ten trials for each condition, and the average was calculated accordingly.

[4]https://github.com/MacTuitui/toio_processing
[5]https://toio.github.io/toio-spec/en/docs/ble_motor/
[6]https://github.com/AxLab-UofC/CHI2023_AeroRigUI
[7]https://toio.github.io/toio-spec/en/docs/ble_motor/#motor-speed-command-values

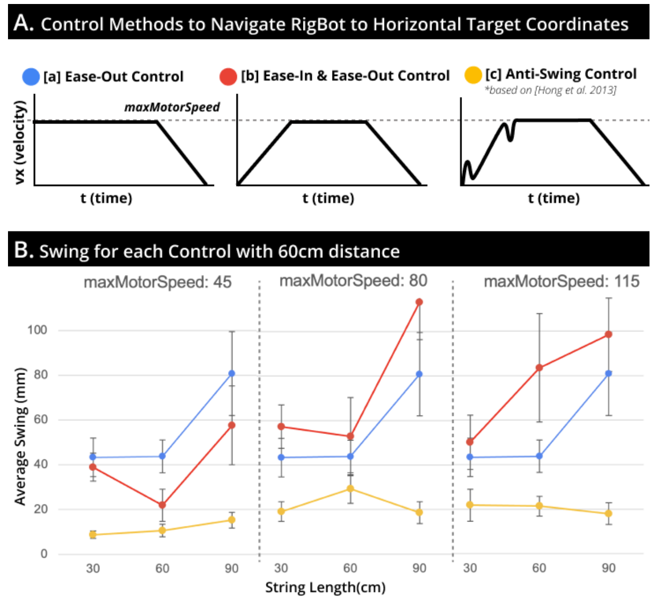


Figure 10: (A) Control methods to navigate RigBot to target XY positions, (B) evaluation results for swing vs. "maxMotorSpeed" parameter for each control method.

The results are shown in Figure 10 B. The [c] Anti-Swing Control, adapted from [21], successfully minimized the swing over other methods in each conditions. On the other hand, we found a trade-off that the time it took for a RigBot to arrive at the target location was slightly longer for [c] compared to others with a scale of 1.15 to 1.5. We incorporated this trade-off into our control system, where users can select different RigBot control modes depending on what they prioritize, either faster speed or less swing.

6.2.3 Summary of Swing Control. In summary, our study has revealed the importance and applicability of control methods to minimize the swing amount, while still, further research could investigate other control methods. Additionally, as swing could also be generated through user interaction, future systems could include a 3D tracking system that allows users to cancel any swinging via an advanced closed-loop control.

6.3 DoF Control Accuracy

Finally, the control accuracy for all 6 degrees of freedom (DoF) of AeroRigUI must be understood to provide usability and controllability of our current prototype. As such, we have conducted a technical evaluation to assess the 6DoF control accuracy. In this section, we outline our evaluation and discuss the results. The evaluation method and result are each depicted in Figure 11 and 12.

6.3.1 3 DoF (x, y, z) Control with a Single RigBot. A single RigBot can move an object in the x, y, and z directions as covered in our design space. We tested the accuracy of a single RigBot's XYZ translation of the rigged object by moving an object to a fixed target position in the GUI from varied origin positions, then comparing it to the actual x, y, and z positions of the object. We measured the ground truth x-y position by marking the RigBot's position on

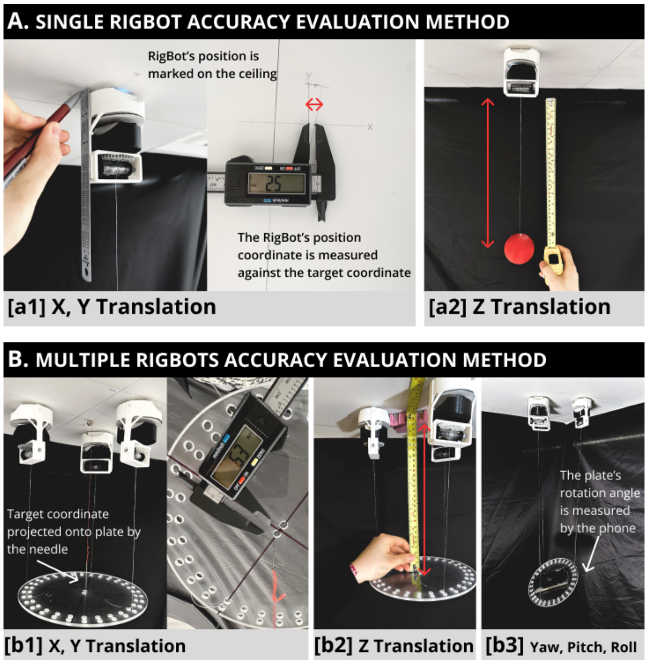
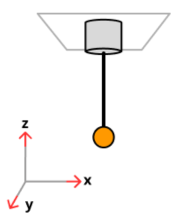


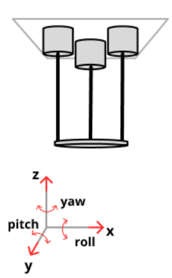
Figure 11: RigBot 6DoF control accuracy evaluation methods for (A) Single RigBot and (B) Multiple RigBots.

A. 3DoF CONTROL ACCURACY FOR A SINGLE RIGBOT



	Average Error	Standard Deviation
X Translation	1.19mm	0.51mm
Y Translation	2.17mm	0.91mm
Z Translation	8.00mm	5.35mm

B. 6DoF CONTROL ACCURACY FOR MULTIPLE RIGBOTS



	Average Error	Standard Deviation
X Translation	3.73mm	1.06mm
Y Translation	2.60mm	1.42mm
Z Translation	12.85mm	2.85mm
Yaw Rotation	1.95°	1.05°
Pitch Rotation	3.50°	1.48°
Roll Rotation	2.45°	2.19°

Figure 12: Evaluation results for the control accuracy of (A) 3DoF motion for a single RigBot, and (B) 6DoF motion for multiple RigBots (study was conducted with 3 RigBots).

the ceiling (Figure 11 [a1]), and measuring the x and y distance to the original coordinate with a caliper. We measured the z position using a tape measure (Figure 11 [a2]). We define the error as the absolute difference between the target coordinate and the actual coordinate for each direction, with the unit of mm. Figure 12A shows our resulting average error and standard deviation from running 20 trials. The average error for translation in the x-direction is 1.19mm with a standard deviation of 0.51mm, for the y-direction, 2.17mm and 0.91mm, and for z-direction, 8mm and 5.35mm. The larger average error for z indicates the challenge of improving the accuracy in controlling string lengths, which could be further addressed in the future.

6.3.2 6 DoF Control (x, y, z, yaw, pitch, roll) with Multiple RigBots. For multiple RigBots, we tested the control accuracy for X, Y, and Z translations and Yaw, Pitch, and Roll translations. For both studies, we utilized three RigBots to rig the acrylic plate used in section 6.1.4, keeping the strings vertical for rigging, which resulted in keeping the rigging point for each RigBot equidistant apart for 15 cm.

For XYZ translation accuracy study, we measured the ground truth x-y position by pointing a needle from the target coordinate onto the rigged acrylic plate and measuring the difference to the center point with a caliper (Figure 11 [b1]). We measured the z position using a tape measure (Figure 11 [b2]). Our results show that the average errors for X, Y, and Z are 3.73mm, 2.80mm, and 12.85mm, respectively. They have standard deviations of 1.06mm, 1.42mm, and 2.85mm, respectively. Overall, the average error for all three directions is slightly higher than single robot control.

For Yaw, Pitch, and Roll translation accuracy, we derived the error by taking the absolute difference in degrees between the randomized target rotation angles in the GUI and the actual rotation angle. We measured the ground truth orientation by attaching a smartphone to the acrylic plate and using the built-in protractor and compass apps to collect data (Figure 11 [b3]). We found the average errors for yaw, pitch, and roll to be 1.95, 3.50, and 2.45 degrees, respectively, and the standard deviations to be 1.05, 1.48, and 2.19.

6.3.3 Summary for DoF Control Accuracy. Our study demonstrated our prototype of AeroRigUI is generally capable of controlling 6DoF control with errors of < 4 mm for XY transition, < 13 mm for Z, and < 4 degrees for Yaw, Pitch, and Roll. In the future, our system can be improved for control accuracy in the z-direction by developing a closed-loop control with a tracking system for the object's height.

7 APPLICATIONS

We present three application areas to highlight the potential use cases within the given architecture and design space of AeroRigUI. Along with demonstrating the applications, we also discuss the technical challenges derived from our technical evaluation (section 6) to elaborate on the feasibility.

7.1 Room Configuration and Control

For the application of room configuration, we demonstrate a collaborative office scenario with the integration of AeroRigUI on the office ceiling that facilitates everyday interactions. To begin, the reconfigurable surface divides the office space into separate spaces for small-group collaboration or individual work (Figure 13a). AeroRigUI also moves and rotates office objects such as notebooks, boards, pen holders, and paper to facilitate the brainstorming process tangibly and physically. As a prior study found that physical post-it notes enhance idea generation [25], AeroRigUI has the potential to organize ideas and notes in 3D space physically, facilitatedby RigBots. Then, the divider reconfigures into a projection screen for an office-wide meeting (Figure 13b). RigBots can tilt and adjust the screen's orientation when interacting with specific individuals (Figure 13c). While the idea boards and projection screens in AeroRigUI are not stably fixed due to their rigging mechanics (not ideal for sketching surfaces), the stability could be improved by rigging with multiple strings. Furthermore, if a future system can better handle and control the swing (e.g. with advanced closed-loop control), such a system could utilize swing, for example, to make some post-it notes appealing with swinging motions.

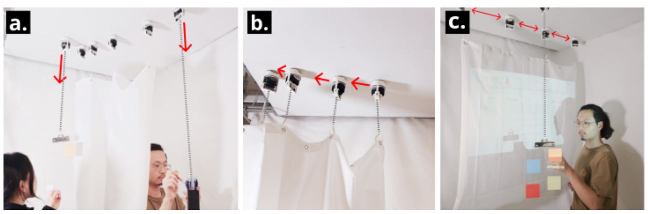


Figure 13: Reconfigurable office space for multi-user interaction (a. aerially-rigged space divider, brainstorm board, and pen holder for collaborative brainstorming; b. RigBots transitioning the reconfigurable surface for different use; c. a rigging projection screen for whole group meeting.)

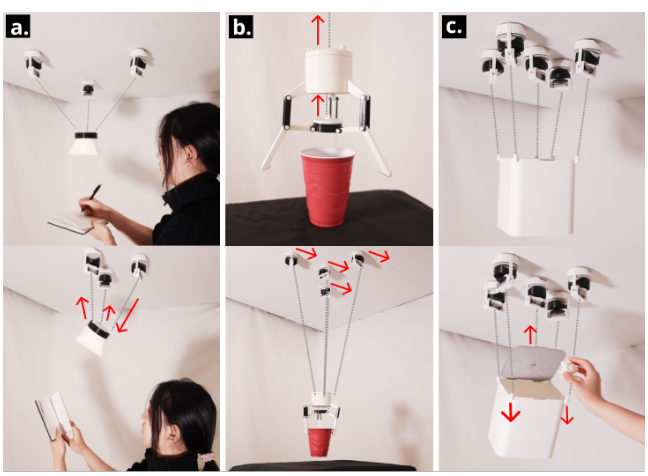


Figure 14: Examples of dynamic room fixtures in everyday space (a. movable light, b. gripper transporting a cup, c. trash container).

Additionally, AeroRigUI allows everyday room fixtures and physical objects to be controlled spatially. For example, a ceiling light controlled by RigBots could follow the user as they move around and/or tilt its angle to shed light on certain areas better to navigate them, utilizing 6DoF control interactively (Figure 14a) [1]. A rigged gripper can move a user's cup to remind them to drink water (Figure 14b). Further, an aerial trash container could approach to a user and open the lid to allow for convenient mobile trashing (Figure 14b). After the trash is thrown into the trash container, it could even automatically transfer the trash to a larger trash bin by tilting it. As both the gripper and trash can are designed to handle extra weight, for safety, the system shall be operated with consideration to prevent RigBot from dropping by exceeding the payload capacity. Hence, the future design could include a load weight detection mechanism.

While everyday floors or tabletop surfaces can often be difficult to navigate for robotic systems due to clutters of obstacles, AeroRigUI, as demonstrated above, enables the augmentation of our living space by fully utilizing underused ceiling surfaces.

7.2 Interactive Astral Representations

To demonstrate how AeroRigUI leverages its 3D spatial object control capability, we present a data representation application with fully spatial astronomical 3D data points. For this application area, we present two displays, the first one displaying **the solar system**, and the second one displaying **the constellation star map**. The solar system demo represents the constantly moving planetary orbits, while the constellation map demonstrates fluid switching from one configuration to another. with these displays, we demonstrate AeroRigUI's great potential to represent spatial and abstract information in physical and tangible ways. In the future, we envision AeroRigUI displays as effective and attractive museum exhibits and classroom displays.

Figure 15 shows our prototype of an aerially-rigged solar system. This demonstration can simulate and replay not only the motion of heliocentric planet motions (orbit with the sun as the center, Figure 15a top), but also the geocentric (orbit with the earth as the center, Figure 15a bottom) orbital motions of the solar system by dynamically reconfiguring. These physical orbiting displays can be manipulated, for example, with another rigged object that acts as a slider input (to control the speed of animation) or a 3D joystick (to control the 3D coordinate position of the solar system), depending on the string constraints (Figure 15b, c). AeroRigUI's spatial representation capability helps physicalize the perspective shifts between the multiple models and helps users understand how humans historically interpreted the universe. For the implementation, as shown in Figure 15b, the camera was mounted above the user's control area to detect the hand's movement between the tangible object and the controller input. One feasibility challenge is in minimizing swing, as our anti-swing control was not designed for constantly moving targets. This could be improved in the future with an advanced control method.

As for the constellation demo, we physically display constellation patterns in 3D, where individual stars are represented with rigged physical objects (shown in Figure 16). While constellations are commonly represented in 2D on screens or printed media, 3D representations with physical objects could convey extra spatial information to deepen the understanding of the star map [38]. Our

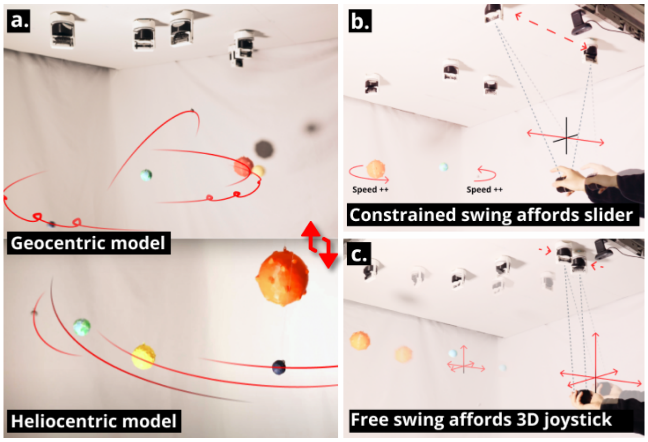


Figure 15: Examples of Solar Model Representations (a. heliocentric vs. geocentric models of the solar-system orbit b. user using a mid-air slider to adjust orbit speed, c. user using a mid-air joystick to translate the model spatially).

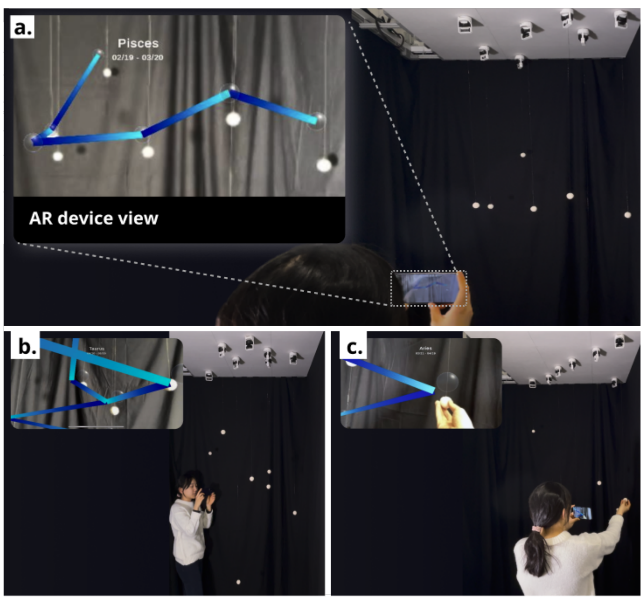


Figure 16: Constellation Application (a. AR application, overlaying graphical information onto physical objects, b. a user walking through aerially rigged objects to explore the data, c. a user touching the physically represented star).

demonstration further coupled the AeroRigUI's actuated objects with overlaid AR information on a smartphone (Figure 16a). While users can interact with the essential data of a 3D star map of each constellation through tangible objects, additional information, such as connecting lines or constellation names, could be overlaid with the AR interface, building on top of an approach in AR and robots [63]. Figure 16 b shows a user walking through the aerially rigged stars to spatially understand where the stars are located with respect to each other. Also, users may learn more about each star through touch and the AR app (Figure 16 c). For the implementation, we prototyped this system using Unity + AR Foundation [8] to build the application. We utilized AR Foundation Remote [9], a tool to run the AR app on iPhones via Mac OS.

7.3 Interactive Animated Craft and Kinetic Sculptures

Lastly, we demonstrate an application in mid-air animated craft and kinetic sculpture that people can interact with (Figure 17a). With hand tracking, the system can map users' finger movements to the rigged objects' motions to enable puppeteering interactions [42]. One example of a rigged, animated character is a flying bird (Figure 17b). The yaw, pitch, and roll of the user's hand can be bonded to the bird's 6DoF motion and flapping motion. Another example is a rigged marionette, whose limbs and head can be individually controlled with fingers (Figure 17c). While hand-based interaction is one of the many potential methods to manipulate and interact with rigged characters, other forms of controlling expressive objects in mid-air can bring novel expressibility. This application is inspired by the aerially rigging technique practiced for theatrical stages [46, 49], and we believe it can be extended for different utilities in interactive kinetic art, entertainment, and human-robot interaction.

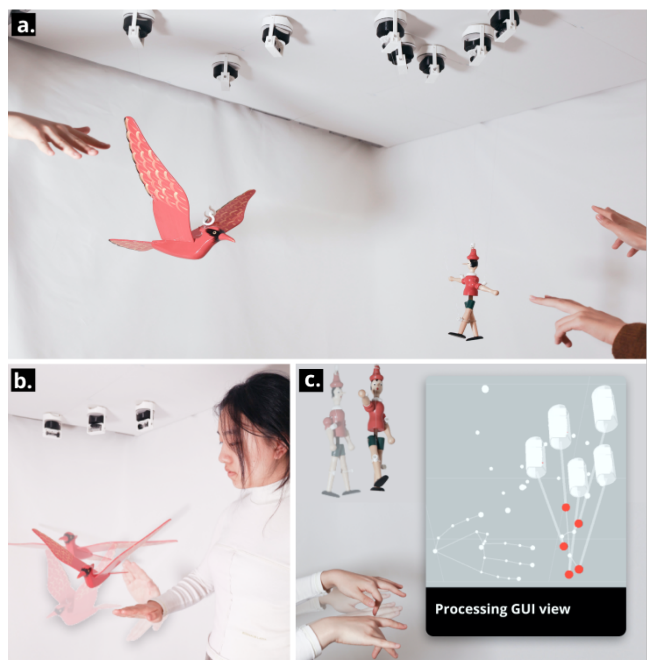


Figure 17: Examples of interactive animated crafts: (a) interactive aerial puppets gesturally controlled by two users' hands, (b) a user making a puppet bird "fly" by tilting their hand, (c) a user moving fingers to control the marionette.

[8] https://unity.com/unity/features/arfoundation
[9] https://assetstore.unity.com/packages/tools/utilities/ar-foundation-remote-2-0-201106

8 LIMITATIONS AND FUTURE WORK

This paper investigated the first step towards enhancing our physical space with ceiling robots rigging physical objects for enriching spatial and tangible interaction. To bring this research further, there is a range of limitations and challenges that we would like to share as future research opportunities.

8.1 General Limitations and Challenges in using Rigging Mechanisms

While the string-based rigging mechanism brings unique benefits for spatial interaction, during our implementation process of the prototype, we encountered difficulties related to rigging.

Swinging: While we have minimized the swing generated with the robot's motion by introducing an Anti-Swing control method ([21]), swings could be further mitigated by applying other swing-elimination techniques [39, 39, 61]. Hardware modifications with damping materials could help eliminate swings. In addition, the system could also cancel the swing generated by external forces, such as human interaction, by tracking the swing of the rigged objects in 3D and adjusting robot motion in an advanced closed-loop control. Furthermore, our research could take advantage of the swing by intentionally generating swings with desired frequency and magnitude to create haptic feedback with impacts.

String Entanglement: String entanglement is another issue that our approach may encounter when multiple robots are moving with an excessive swing. Though minimizing swing can reduce the chance of entanglement, hardware modification, for example, strings with less friction, could be another way to mitigate entanglement.

Safety: As discussed in the technical evaluation, safety is a crucial factor to be addressed since a RigBot can drop from the ceiling in certain conditions. In addition, to further improve safety and maximize payload, we could implement hardware changes, such as optimized wheels with strong, attractive magnetic force to the ceiling or rails in a grid on the ceiling for RigBots to travel along. Additionally, to avoid any injury or hardware damage in unforeseen circumstances, covering the hardware with soft cushioning materials should be considered for practical usage.

8.2 Hardware Design

General Robot Design and String: While we employed toio robots to demonstrate the concept of AeroRigUI, developing custom-designed robots could improve the system's capability. For example, if we could use smaller robots, we could reduce the minimum gap between aerially rigged objects to improve the resolution of mid-air physical display applications. Similarly, selecting or engineering RigBot properties (e.g. speed, lifting force, sizes, etc.) would be an essential practice for future designers and researchers to build interactive applications that satisfy their goals.

Exploration of different string types (elasticity, stiffness) and how they affect properties such as stability and haptics is another direction we could take. Future research could even utilize actuated fiber materials for reeling strings to enrich haptic interactions and object movement expression [28].

Improving Reconfigurability: While the toio robots were embedded in our RigBot implementation, future implementation could employ a docking and undocking system, similar to HERMITS [40]. In such a way, the core self-propelled robotic component, for example, toios, may dock and undock from different passive mechanical shells that have different types of strings and connectors attached to them. This could increase the number of rigged objects a single (or fewer) robot can control, reducing the cost of the system as well.

Furthermore, the reconfigurability feature could be implemented to allow RigBots to attach and detach to different objects for control flexibly. Such a feature could be built by updating our current design, where we used magnet connectors or hooks to change rigging objects manually.

Battery and Charging: Currently, we have observed that our system is capable of more than two hours of operation with constant actuation thanks to the toio hardware. Future systems could incorporate self-charging stations on the ceiling that allow wireless charging for idle robots to maximize robot working time on the ceiling without manual intervention.

Ceiling Surfaces: Our ceiling surface implementation has a relatively simple design with a combination of a ferromagnetic metal sheet and a tracking mat for toio. As the available toio mat is limited to a dimension of 1260mm width and 1188mm length, the future prototype could overcome the scale limit by introducing scalable tracking methods. Additionally, as we envision our system to be integrated into everyday space / ceilings, improved methods to easily turn any ceiling surface into an AeroRigUI-ready surface could be explored. Ideal methods are those integrated into traditional ways we customize our living space, for example, by pasting wallpapers.

Towards Allowing for Robust Tangible Interaction: While the current system has essential limitations from users' external forces failing the system that users have to be careful when interacting with the system, we believe there are approaches to building a system to allow for robust, tangible interaction. One promising approach is to learn from SPIDAR [52], where the reeling mechanism is back-drivable. Additionally, future RigBot hardware could incorporate extra actuators to increase the magnetic attraction when the device expects tangible interaction by users. In such a way, the RigBots can still travel across the surface when the magnetic attraction is switched off. As physically actuated spatial UI systems are inherent with affordance for tangible interaction, these technical efforts shall bring our system to allow for robust tangible and embodied interaction.

8.3 Software Control

There are several opportunities in software control as well. For example, some other advanced control could be implemented to provide closed-loop haptic feedback that could provide users with variable force or compliance feedback [52]. We wish to improve our GUI further to make it accessible and easy to use for a broad set of users to design interaction and display with AeroRigUI.

8.4 Further Applications and User Study

Beyond the applications presented in our paper, AeroRigUI has a broader space for applications when combined with other user interface modalities. For example, for VR and AR domains (where room-scaled haptics have been investigated in recent years [5, 24, 62, 64, 66]), AeroRigUI can move haptic props from the ceiling with full control of objects in 3D space. This could be a relatively feasible and cost-efficient way to provide room-scale haptics. In this case, it is important to develop advanced control, for example, where mid-air haptic props can be actuated in response to users' hands reaching out.

Another future application is kinesthetic learning(e.g. learning motor skills), which often involves spatial understanding of people's bodies [56]. We believe AeroRigUI can tangibly support such bodily applications as our preliminary exploration shown in Figure 18, for example, to assist a badminton racket's 6DoF motion for novice player (a), or for soccer players to practice making contact with a ball from any height (b). While the current implementation has difficulty stably supporting this application due to the payload capacity limitation, it would be an exciting direction with future prototypes.

Figure 18: Our preliminary exploration for kinesthetic learning application: (a) rigged badminton racket with motion playback, (b) soccer practice by leveraging height affordances.

Other everyday living and working scenarios can be a great space to explore, including dining/cooking experience, factory automation, as well as 3D material fabrication/crafts, where advanced object tracking and control may need to be incorporated.

To investigate more interactive applications, user evaluations of how people interact, interpret and perceive rigged physical objects should be conducted. Specifically, we could evaluate the effectiveness of height affordances and the string-based dynamic haptic constraints on users in everyday circumstances. We would also like to conduct a workshop to allow designers, artists, or even children to develop their custom applications with our system to test our GUI and system capabilities further.

8.5 Enriching Space by Coordinating Robots on Floor, Ceiling, and Wall

Lastly, we ambitiously envision a future where RigBots on ceiling surfaces can be coordinated with other mobile robots in the physical space, either on the floor, table, or wall. While rigging objects only from the ceilings cause limitations, such as the directionality of force, this future system could dynamically rig and control physical objects from different surfaces in space [22, 41]. While there are a lot of technical challenges, orchestrating robotic systems on different room surfaces can be a grand approach to enriching our physical space dynamically, reconfigurably, and adaptively.

9 CONCLUSION

In this paper, we introduced AeroRigUI, a novel approach for designing mid-air spatial tangible experiences based on mobile swarm robots with reeling mechanisms. We introduced a modular hardware system with an overhanging ceiling structure, and shell and magnet modifications to off-the-shelf mobile toio robots. We explored different rigging methods and how this approach introduces various types of actuation, control, and affordance for interaction design. Technical evaluations were introduced to analyze and access the technical challenge for the system, including payload capacity and swing. A range of potential applications was demonstrated with our implementation, including 3D data physicalization, interactive animated craft and kinetic sculptures, and room reconfiguration. In all, we hope our work inspires future research on the enrichment of physical space with mobile robots on room surfaces for advanced deployability and controllability.

ACKNOWLEDGMENTS

We thank Timmy Lin for his assistance in building the ceiling structure and helping with the video. We also appreciate Emilie Faracci for photography help, and Marco Wang for assistance in filming and coding. Lastly, we thank Anup Sathya for proof-reading and members of the Actuated Experience Lab for their support.

REFERENCES

[1] Harshit Agrawal, Sang-won Leigh, and Pattie Maes. 2015. L'evolved: autonomous and ubiquitous utilities as smart agents. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 487–491.
[2] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. 2018. Grand challenges in shape-changing interface research. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–14.
[3] Tobias Alrøe, Jonas Grann, Erik Grönvall, Marianne Graves Petersen, and Jesper L Rasmussen. 2012. Aerial tunes: exploring interaction qualities of mid-air displays. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design*. 514–523.
[4] ART+COM. 2008. *Kinetic Sculpture - the Shape of Things to Come.* Retrieved September 15, 2022 from https://artcom.de/en/?project=kinetic-sculpture
[5] Elodie Bouzbib, Gilles Bailly, Sinan Haliyo, and Pascal Frey. 2020. CoVR: A large-scale force-feedback robotic interface for non-deterministic scenarios in vr. *arXiv preprint arXiv:2009.07149* (2020).
[6] Sean Braley, Calvin Rubens, Timothy Merritt, and Roel Vertegaal. 2018. GridDrones: A Self-Levitating Physical Voxel Lattice for Interactive 3D Surface Deformations.. In *UIST*, Vol. 18. D200.
[7] Laroussi Buoguila, Masahiro Ishii, and Makoto Sato. 2000. Scaleable spidar: A haptic interface for human-scale virtual environments. In *International Workshop on Haptic Human-Computer Interaction*. Springer, 182–193.
[8] Alexandre Campeau-Lecours, Simon Foucault, Thierry Laliberte, Boris Mayer-St-Onge, and Clement Gosselin. 2016. A cable-suspended intelligent crane assist device for the intuitive manipulation of large payloads. *IEEE/ASME Transactions on Mechatronics* 21, 4 (2016), 2073–2084.
[9] Tom Carter, Sue Ann Seah, Benjamin Long, Bruce Drinkwater, and Sriram Subramanian. 2013. UltraHaptics: multi-point mid-air haptic feedback for touch surfaces. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. 505–514.
[10] CircusArts. 2017. *Sailor Circus Duo Trapeze Performance at 2017 Smithsonian Folklife Festival.* Retrieved September 15, 2022 from https://www.youtube.com/watch?v=XgWRj4LB6nM&ab_channel=CircusArts
[11] Marcelo Coelho and Jamie Zigelbaum. 2011. Shape-changing interfaces. *Personal and Ubiquitous Computing* 15, 2 (2011), 161–173.
[12] Severin Engert, Konstantin Klamka, Andreas Peetz, and Raimund Dachselt. 2022. STRAIDE: A Research Platform for Shape-Changing Spatial Displays based on Actuated Strings. In *CHI Conference on Human Factors in Computing Systems*. 1–16.
[13] Sony Interactive Entertainment. 2020. *toio Mat for Developers.* Retrieved May 16, 2022 from https://toio.io/news/2020/20200423-1.html

[14] Cathy Fang, Yang Zhang, Matthew Dworman, and Chris Harrison. 2020. Wireality: Enabling complex tangible geometries in virtual reality with worn multi-string haptics. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–10.

[15] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: dynamic physical affordances and constraints through shape and object actuation.. In *Uist*, Vol. 13. 2501–988.

[16] Masahiro Fujita, Suguru Ikeda, Toshiaki Fujimoto, Toshihiko Shimizu, Shuhei Ikemoto, and Takeshi Miyamoto. 2018. Development of universal vacuum gripper for wall-climbing robot. *Advanced Robotics* 32, 6 (2018), 283–296.

[17] Rui Fukui, Hiroshi Morishita, Taketoshi Mori, and Tomomasa Sato. 2011. Hangbot: A ceiling mobile robot with robust locomotion under a large payload (key mechanisms integration and performance experiments). In *2011 IEEE International Conference on Robotics and Automation*. IEEE, 4601–4607.

[18] Rui Fukui, Hiroshi Morishita, and Tomomasa Sato. 2014. Experimental comparison of two ceiling hanging mobile robots through real prototypes development. *Journal of Robotics and Mechatronics* 26, 1 (2014), 40–50.

[19] Takuro Furumoto, Takumi Kasai, Masahiro Fujiwara, Yasutoshi Makino, and Hiroyuki Shinoda. 2021. Midair Balloon Interface: A Soft and Lightweight Midair Object for Proximate Interactions. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 783–795.

[20] Ryuji Hirayama, Diego Martinez Plasencia, Nobuyuki Masuda, and Sriram Subramanian. 2019. A volumetric display for visual, tactile and audio presentation using acoustic trapping. *Nature* 575, 7782 (2019), 320–323.

[21] Chaofei Hong, Gening Xu, and Yuanfang Tao. 2013. A Novel Optimal Trajectory Planning for Overhead Traveling Crane. *Researches and Applications in Mechanical Engineering* 2, 4 (2013), 75–80.

[22] Junyu Hu, Xu Han, Yourui Tao, and Shizhe Feng. 2022. A magnetic crawler wall-climbing robot with capacity of high payload on the convex surface. *Robotics and Autonomous Systems* 148 (2022), 103907.

[23] Anusha Jayasiri, Katsuhito Akahane, and Makoto Sato. 2012. Haptic rendering of dynamic image sequence using string based haptic device SPIDAR. (2012).

[24] Seungwoo Je, Hyunseung Lim, Kongpyung Moon, Shan-Yuan Teng, Jas Brooks, Pedro Lopes, and Andrea Bianchi. 2021. Elevate: A walkable pin-array for large shape-changing terrains. In *Proceedings of the 2021 CHI Conference on human Factors in Computing Systems*. 1–11.

[25] Mads Møller Jensen, Sarah-Kristin Thiel, Eve Hoggan, and Susanne Bødker. 2018. Physical versus digital sticky notes in collaborative ideation. *Computer Supported Cooperative Work (CSCW)* 27, 3 (2018), 609–645.

[26] Elizabeth Jochum, Jarvis Schultz, Elliot Johnson, and Todd D Murphey. 2014. Robotic puppets and the engineering of autonomous theater. In *Controls and Art*. Springer, 107–128.

[27] Elliot Johnson and Todd D Murphey. 2007. Dynamic modeling and motion planning for marionettes: Rigid bodies articulated by massless strings. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 330–335.

[28] Ozgun Kilic Afsar, Ali Shtarbanov, Hila Mor, Ken Nakagaki, Jack Forman, Karen Modrei, Seung Hee Jeong, Klas Hjort, Kristina Höök, and Hiroshi Ishii. 2021. OmniFiber: Integrated Fluidic Fiber Actuators for Weaving Movement Based Interactions into the 'Fabric of Everyday Life'. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 1010–1026.

[29] Lawrence H Kim and Sean Follmer. 2017. Ubiswarm: Ubiquitous robotic interfaces and investigation of abstract motion as a display. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–20.

[30] Lawrence H Kim and Sean Follmer. 2019. Swarmhaptics: Haptic display with swarm robots. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–13.

[31] Sangbae Kim, Matthew Spenko, Salomon Trujillo, Barrett Heyneman, Virgilio Mattoli, and Mark R Cutkosky. 2007. Whole body adhesion: hierarchical, directional and distributed control of adhesive forces for a climbing robot. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 1268–1273.

[32] Ryo Kishi. 2015. *atOms*. Retrieved September 15, 2022 from http://ryokishi.org/works/atoms21/atoms21.html

[33] Idina Menzel Kristin Chenoweth. 2004. *Idina & Kristin - Tony's 2004 - Defying Gravity*. Retrieved May 26, 2022 from https://www.youtube.com/watch?v=O5V9KwppMfs&ab_channel=AGillingham

[34] Steven LaValle. 2006. *Yaw, pitch, and roll rotations*. Retrieved September 15, 2022 from http://planning.cs.uiuc.edu/node102.html

[35] Mathieu Le Goc, Lawrence H Kim, Ali Parsaei, Jean-Daniel Fekete, Pierre Dragicevic, and Sean Follmer. 2016. Zooids: Building blocks for swarm user interfaces. In *Proceedings of the 29th annual symposium on user interface software and technology*. 97–109.

[36] Ho-Hoon Lee. 1998. Modeling and control of a three-dimensional overhead crane. (1998).

[37] Jinha Lee, Rehmi Post, and Hiroshi Ishii. 2011. ZeroN: mid-air tangible interaction enabled by computer controlled magnetic levitation. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. 327–336.

[38] Jiatong Li, Chenfeng Gao, and Ken Nakagaki. 2022. ShadowAstro: Levitating Constellation Silhouette for Spatial Exploration and Learning. In *The Adjunct Publication of the 35th Annual ACM Symposium on User Interface Software and Technology*. 1–3.

[39] Y Miao, F Xu, Y Hu, J An, and M Zhang. 2019. Anti-swing control of the overhead crane system based on the harmony search radial basis function neural network algorithm. (2019). https://doi.org/10.1177/1687814019834458

[40] Ken Nakagaki, Joanne Leong, Jordan L Tappa, João Wilbert, and Hiroshi Ishii. 2020. Hermits: Dynamically reconfiguring the interactivity of self-propelled tuis with mechanical shell add-ons. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 882–896.

[41] Ken Nakagaki, Jordan L Tappa, Yi Zheng, Jack Forman, Joanne Leong, Sven Koenig, and Hiroshi Ishii. 2022. (Dis) Appearables: A Concept and Method for Actuated Tangible UIs to Appear and Disappear based on Stages. In *CHI Conference on Human Factors in Computing Systems*. 1–13.

[42] Ken Nakagaki, Udayan Umapathi, Daniel Leithinger, and Hiroshi Ishii. 2017. AnimaStage: hands-on animated craft on pin-based shape displays. In *Proceedings of the 2017 Conference on Designing Interactive Systems*. 1093–1097.

[43] Yoichi Ochiai, Takayuki Hoshi, and Jun Rekimoto. 2014. Pixie dust: graphics generated by levitated and animated objects in computational acoustic-potential field. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–13.

[44] Yoichi Ochiai, Takayuki Hoshi, and Jun Rekimoto. 2014. Three-dimensional mid-air acoustic manipulation by ultrasonic phased arrays. *PloS one* 9, 5 (2014), e97590.

[45] DCD Oguamanam, JS Hansen, and GR Heppler. 2001. Dynamics of a three-dimensional overhead crane system. *Journal of sound and vibration* 242, 3 (2001), 411–426.

[46] Kalpesh B Pathak and Ketan K Acharya. 2011. Automation in entertainment industry. In *2011 Nirma University International Conference on Engineering*. IEEE, 1–6.

[47] Ivan Poupyrev, Tatsushi Nashida, and Makoto Okabe. 2007. Actuation and tangible user interfaces: the Vaucanson duck, robots, and shape displays. In *Proceedings of the 1st international conference on Tangible and embedded interaction*. 205–212.

[48] MK Rasmussen, EW Pedersen, MG Petersen, and K Hornbæk. 2012. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. (2012).

[49] Guy Richards. 2008. Stages of automation-[Control theatre]. *Engineering & Technology* 3, 21 (2008), 38–41.

[50] A Roudaut, A Karnik, and M Löchtefeld. 2013. Subramanian,. S. Morphees: Toward High" Shape Resolution. *Self-actuated Flexible Mobile Devices. In Proc. CHI* 13 (2013), 593–602.

[51] Calvin Rubens, Sean Braley, Antonio Gomes, Daniel Goc, Xujing Zhang, Juan Pablo Carrascal, and Roel Vertegaal. 2015. Bitdrones: Towards levitating programmable matter using interactive 3d quadcopter displays. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 57–58.

[52] Makoto Sato. 2002. Development of string-based force display: SPIDAR. In *8th international conference on virtual systems and multimedia*. Citeseer, 1034–1039.

[53] Makoto Sato. 2002. SPIDAR and virtual reality. In *Proceedings of the 5th Biannual World Automation Congress*, Vol. 13. IEEE, 17–23.

[54] Tomomasa Sato, Rui Fukui, Hiroshi Morishita, and Taketoshi Mori. 2004. Construction of ceiling adsorbed mobile robots platform utilizing permanent magnet inductive traction method. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, Vol. 1. IEEE, 552–558.

[55] Vivian Shen, Craig Shultz, and Chris Harrison. 2022. Mouth Haptics in VR using a Headset Ultrasound Phased Array. In *CHI Conference on Human Factors in Computing Systems*. 1–14.

[56] Kirin Sinha. 2014. *Kinesthetic Learning: Moving Toward a New Model for Education*. Retrieved September 15, 2022 from https://www.edutopia.org/blog/kinesthetic-learning-new-model-education-kirin-sinha

[57] Aaron Sirken, Gedaliah Knizhnik, Jessica McWilliams, and Sarah Bergbreiter. 2017. Bridge risk investigation diagnostic grouped exploratory (BRIDGE) bot. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 6526–6532.

[58] Rajinder Sodhi, Ivan Poupyrev, Matthew Glisson, and Ali Israr. 2013. AIREAL: interactive tactile experiences in free air. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.

[59] Gabor Stepan, Andras Toth, Laszlo Kovacs, Gunnar Bolmsjo, Giorgos Nikoleris, Dragoljub Surdilovic, Andreas Conrad, Antonios Gasteratos, Nikolaos Kyriakoulis, Dimitrios Chrysostomou, et al. 2009. ACROBOTER: a ceiling based crawling, hoisting and swinging service robot platform. In *Beyond gray droids: domestic robot design for the 21st century workshop at HCI*, Vol. 2009. 2.

[60] Sriram Subramanian, Sue Ann Seah, Hiroyuki Shinoda, Eve Hoggan, and Loic Corenthy. 2016. Mid-air haptics and displays: systems for un-instrumented mid-air interactions. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. 3446–3452.

[61] Ning Sun and Yongchun Fang. 2012. New Energy Analytical Results for the Regulation of Underactuated Overhead Cranes: An End-Effector Motion-Based

Approach. *IEEE Transactions on Industrial Electronics* 59, 12 (2012), 4723–4734. https://doi.org/10.1109/TIE.2012.2183837

[62] Ryo Suzuki, Hooman Hedayati, Clement Zheng, James L Bohn, Daniel Szafir, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2020. Roomshift: Room-scale dynamic haptics for vr with furniture-moving swarm robots. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–11.

[63] Ryo Suzuki, Adnan Karim, Tian Xia, Hooman Hedayati, and Nicolai Marquardt. 2022. Augmented Reality and Robotics: A Survey and Taxonomy for AR-enhanced Human-Robot Interaction and Robotic Interfaces. In *CHI Conference on Human Factors in Computing Systems*. 1–33.

[64] Ryo Suzuki, Ryosuke Nakayama, Dan Liu, Yasuaki Kakehi, Mark D Gross, and Daniel Leithinger. 2020. LiftTiles: constructive building blocks for prototyping room-scale shape-changing interfaces. In *Proceedings of the fourteenth international conference on tangible, embedded, and embodied interaction*. 143–151.

[65] Ryo Suzuki, Eyal Ofek, Mike Sinclair, Daniel Leithinger, and Mar Gonzalez-Franco. 2021. HapticBots: Distributed Encountered-type Haptics for VR with Multiple Shape-changing Mobile Robots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 1269–1281.

[66] Shan-Yuan Teng, Cheng-Lung Lin, Chi-huan Chiang, Tzu-Sheng Kuo, Liwei Chan, Da-Yuan Huang, and Bing-Yu Chen. 2019. Tilepop: Tile-type pop-up prop for virtual reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 639–649.

[67] Wikipedia. 2021. *Aerial rigging*. Retrieved September 15, 2022 from https://en.wikipedia.org/wiki/Aerial_rigging#:~:text=Aerial%20rigging%20is%20the%20process,the%20artists%20and%20the%20audience.

[68] WOW. 2012. *BLOOM SKIN*. https://www.w0w.co.jp/en/works/bloomskin

[69] Junichi Yamaoka. 2014. *Morphing Cube*. https://www.junichiyamaoka.net/posts/1748995?categoryIds=1959603

[70] Zeyu Yan, Anup Sathya, Pedro Carvalho, Yongquan Hu, Annan Li, and Huaishu Peng. 2021. Towards On-the-wall Tangible Interaction: Using Walls as Interactive, Dynamic, and Responsive User Interface. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–6.

[71] Yu Yoshida and Shugen Ma. 2010. Design of a wall-climbing robot with passive suction cups. In *2010 IEEE International Conference on Robotics and Biomimetics*. IEEE, 1513–1518.

[72] Toshiya Yui and Tomoko Hashida. 2016. Floatio: Floating tangible user interface based on animacy perception. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 43–45.

[73] Yiwei Zhao, Lawrence H Kim, Ye Wang, Mathieu Le Goc, and Sean Follmer. 2017. Robotic assembly of haptic proxy objects for tangible interaction and virtual reality. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. 82–91.

[74] Simon Zimmermann, Roi Poranne, James M Bern, and Stelian Coros. 2019. Puppetmaster: robotic animation of marionettes. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–11.