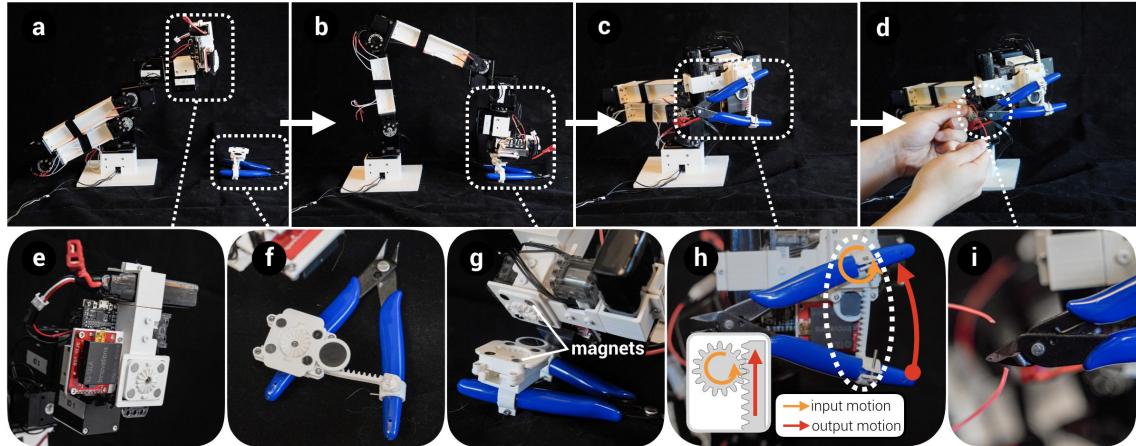


1 Roman: Making Everyday Objects Robotically Manipulable with 3D-Printable
2 Add-on Mechanisms
3

4 ANONYMOUS FOR REVIEW
5
6



8 Fig. 1. Roman is a novel robotic design making everyday handheld objects more robotically manipulable, *i.e.*, easier to be manipulated
9 by a conventional robotic arm. This figure shows a sequence of a Roman-enabled robotic arm picking up a wire cutter on the desk
10 and performing a wire cutting task collaborating with a human (a-d). Roman provides a magnetic gripper (e) for the robotic arm to
11 easily attach and augment the wire cutter with Roman mechanism (f). Snapping in the mechanism using magnets (g), the gripper can
12 actuate the gear-rack movement on the wire cutter (h) to perform the cutting task by squeezing the cutter's handles (i).

13 One important vision of robotics is to provide physical assistance by manipulating different everyday objects, e.g., hand tools, kitchen
14 utensils. However, many objects designed for dexterous hand-control are not easily manipulable by a single robotic arm with a generic
15 parallel gripper. Complementary to existing research on developing grippers and control algorithms, we present Roman, a suite of
16 hardware design and software tool support for robotic engineers to create 3D printable mechanisms attached to everyday handheld
17 objects, making them easier to be manipulated by conventional robotic arms. The Roman hardware comes with a versatile magnetic
18 gripper that can snap on/off handheld objects and drive add-on mechanisms to perform tasks. Roman also provides software support
19 to register and author control programs. To validate our approach, we designed and fabricated Roman mechanisms for 14 everyday
20 objects/tasks presented within a design space and conducted expert interviews with robotic engineers indicating that Roman serves as a
21 practical alternative for enabling robotic manipulation of everyday objects.
22

23
24 CCS Concepts: • Human-centered computing → Interactive systems and tools.
25

26 Additional Key Words and Phrases: Robotic grasping and manipulation; handheld objects augmentation; mechanism design.
27

28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Association for Computing Machinery.

Manuscript submitted to ACM

53 ACM Reference Format:

54 Anonymous for review. 2018. Roman: Making Everyday Objects Robotically Manipulable with 3D-Printable Add-on Mechanisms. In
 55 *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 25 pages.
 56 <https://doi.org/10.1145/1122445.1122456>
 57

58 1 INTRODUCTION

60 Various types of robots that inhabit our living spaces, such as vacuum cleaners, robotic toys, home/office-assistive robots,
 61 promise to aid humans with numerous everyday tasks. Future robots are expected to provide physical assistance for the
 62 elderly at home or help with household chores that require the use of diverse tools and everyday objects [26].
 63

64 Many recent advances in robotic research (primarily on manipulation [17, 37, 52, 53]) aim to solve robotic arms'
 65 manipulation problem for niche, fixed tasks, *e.g.*, opening medicine bottles, handling cookware, and sorting waste.
 66 However, creating a *universally dexterous* robotic arm remains challenging as there are a large number of everyday objects
 67 that are not robotically manipulable, *i.e.*, difficult to manipulate by a consumer-level robotic arm with a generic parallel
 68 gripper, especially so for those *dynamic* objects that have multiple mechanically movable parts, *e.g.*, a pair of pliers and a
 69 spray bottle with a pump pushable toward the body.
 70

71 To tackle this robotic manipulability challenge, prior work often considers the action of the gripper and the manipulator
 72 as completely decoupled, which reduces the complexity of task planning and control involved [3]. To date, researchers
 73 have focused on developing manipulation strategies by analyzing the best grasping points of *static* objects [37, 52].
 74 However, when it comes to *dynamic* objects such as a spray bottle with a pump pushable toward the body, prior work
 75 tends to mimic the pose and force of a human with *multiple* robotic arm [13, 36] or dexterous robotic hand [1] rather than
 76 tackling the task self-containedly with a *single* robotic arm. Meanwhile, on the objects' side, augmenting objects with
 77 actuatable mechanisms is a new approach to enhance their interactivity or functionality (*e.g.*, [8, 33]); however, little has
 78 been explored on how to enable robotic arms to better manipulate such augmented objects.
 79

80 We present Roman—a suite of hardware design and software tool support for robotic engineers to make everyday
 81 handheld objects more Robotically manipulable by a consumer-grade 6-DoF robotic arm.
 82

83 Roman's hardware components consist of (*i*) a library of 3D printable powerless mechanisms that are attached to and
 84 can drive different handheld objects to perform specific tasks, *e.g.*, squeezing a cutter to cut wires (Figure 1a→d); (*ii*) a
 85 gripper (Figure 1e) that uses magnets to securely and automatically attach/detach a robotic arm to/from an object¹. The
 86 gripper also contains an RFID-based module to recognize which object the robotic arm is expected to manipulate thus to
 87 run the corresponding control program for specific tasks.
 88

89 Roman's software component is a user interface for robotic engineers to register and specify the motor input of custom
 90 tasks with pre-defined templates and real-time feedback. A user can rapidly and iteratively author a control program for a
 91 Roman-enabled robotic arm to manipulate a handheld object. Note that Roman does *not* reinvent tools for generating 3D
 92 models of add-on mechanism, which is already supported (*e.g.*, [9, 33]); rather, Roman's software focuses on creating a
 93 control program to operate such mechanisms as different handheld objects require unique sequences of action to perform
 94 a specific task, which would otherwise be tedious to specify even for robotic experts.
 95

96 To validate Roman, we first designed and fabricated mechanisms for 14 everyday objects to demonstrate how a generic
 97 6-DoF robotic arm (built with 3D printed links and Dynamixel XH-540 motors²) can manipulate these objects to perform
 98 specific tasks. Further, we conducted a preliminary interview with four robotic engineers. Participants found Roman's
 99

100 ¹Caveat: currently, an object's placement needs to be known for auto-attachment
 101 ²<https://www.robotis.us/dynamixel-xh540-w150-t/>

105 approach complementary to the current research on robotic manipulation within the human environment, which could be
106 beneficial in specific task scenarios such as cooking, electronics assembly, and caring for house plants. Provided with the
107 pre-fabricated gripper and attaching mechanisms, all participants were able to use Roman's software module to replicate
108 the wire cutter scenario (Figure 1).

110 111 1.1 Contributions

112 Overall, Roman empowers robotics engineers to use 3D printable powerless mechanisms attached to different objects for
113 enhancing these objects' robotic manipulability with task-specific control program embedded in the mechanisms. Our
114 specific contributions are as follows.

- 115 • **Categorization of the robotic manipulability problem in everyday handheld objects (§2)**, including the challenge of speed and range of motion, and the challenge of dexterity when manipulating an dynamic object with its constituent parts (squeezing, twisting, and pumping);
- 116 • **A library of 3D printable add-on mechanisms able to manipulate both static and dynamic objects (§6.2)** based on variations and combinations of spur gears, bevel gears, gear racks, and pin-in-slot mechanisms, which can be attached to everyday objects to enhance their manipulability by a consumer-grade 6-DoF robotic arm;
- 117 • **The design of a versatile magnetic gripper (§6.1)** that can securely attach to the add-on mechanisms on an object, run the control program to drive the mechanism, and detach from the mechanism without outside intervention;
- 118 • **Software that enables robotic engineers to interactively author a program (§7)** to control the Roman mechanism-installed hardware to perform object-specific tasks.

120 121 122 123 124 125 126 127 128 129 1.2 Limitation

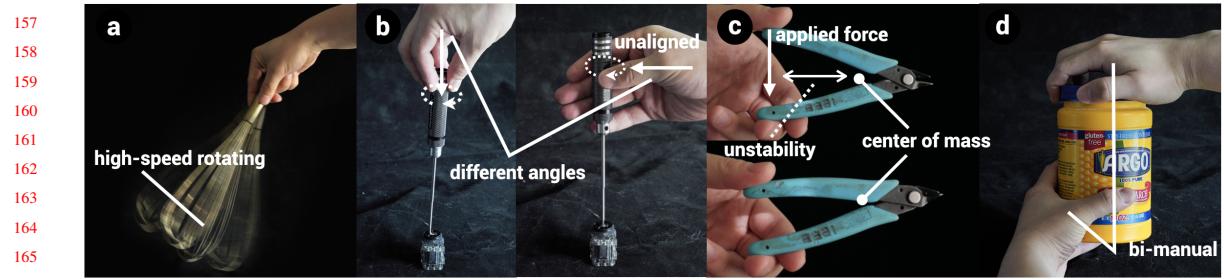
130 At present, Roman has not achieved total autonomy, as there is no sensing modules to detect where an object is positioned or how it is oriented for pick-up (which is a separate topic well studied in robotics and computer vision [14, 18]). Currently, in most of our examples, the object is handed off to the robotic arm by a human (*e.g.*, kitchen utensils) or the position of the object is manually input to the robotic control program by the human user (*e.g.*, the spray bottle example).

137 138 2 CATEGORIZING CHALLENGES OF HANDHELD OBJECTS FOR ROBOTIC MANIPULATION

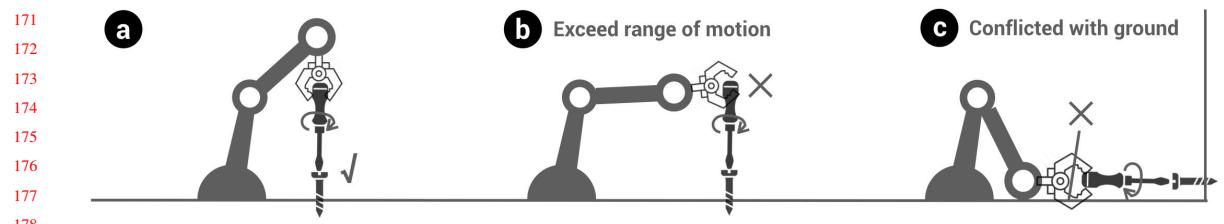
139 Many everyday objects are not robotically manipulable due to the following challenges:

- 140 • **Speed and range of motion challenges** when manipulating an object as a whole. For example, objects that require a high speed/frequency of manipulation may exceed the motor's capability at the end-effector, *e.g.*, a rapidly rotating whisker at a high speed to whip cream (Figure 2a). Meanwhile, manipulating a screwdriver at certain angles might exceed the reach of a situated robotic arm (Figure 2b).
- 141 • **Dexterity-related challenges** when manipulating an object by its constituent parts. For example, to cut a wire, a robotic arm would first need to grasp the cutter legs then apply a steady force to squeeze the handles (Figure 2c); however, a generic parallel gripper might find it hard to both securely grasp *and* squeeze the cutter. Bi-manual manipulation is even harder, *e.g.*, opening a jar lid (Figure 2d) is almost impossible for a single robotic arm and requires either a holding stand or assistance from another robotic arm or a human hand.

142 Our goal is to make everyday **handheld objects** manipulable by a consumer-grade generic 6-DoF robotic arm. To achieve this, we first need to understand *what* objects are currently inaccessible and *how* they are inaccessible for robotic manipulation. Everyday objects are mostly designed to be manipulated by humans. With flexible fingers and



167 Fig. 2. A human hand manipulating a variety of objects that present challenges for robotic arms such as using a whisk (a), rotating a
168 screwdriver (b), squeezing a wire cutter (c), and opening a jar lid (d).



179 Fig. 3. Why manipulating a screwdriver might require range of motion beyond a robotic arm's capability: (a) normal working
180 grasp/distance; (b) when the distance increases, the grasp changes and the end-effector can no longer rotate around the screwdriver's
181 axis; (c) in some edge cases, the manipulation might be interfered by the physical surroundings, e.g., the ground.

183
184 the coordination of four limbs, humans are able to manipulate a wide range of objects with different manipulation
185 complexities, from picking up a cup to playing a piano. In contrast, there exist a large number of everyday objects that
186 are not manipulable by generic robotic arms due to the fact that a general-purpose gripper has limitations in performing
187 different types of grasping and manipulation tasks [3]. As shown in Figure 4, we consider two categories of such objects:
188 those manipulated as a whole (e.g., raising/lowering a knife, rotating a screwdriver) vs. those manipulated by constituent
189 parts (e.g., squeezing a pair of pliers' handles, twisting a pepper grinder cap against the bottle, pushing the pump towards
190 the body of a bottle of hand sanitizer).

194 2.1 Objects Manipulated as a Whole (Static Objects)

195 As shown in Figure 4A-B, objects in this category require a ‘grasp & move’ type of manipulation as the body of the object moves as a whole while performing tasks, e.g., the whole knife moves vertically while chopping vegetables (linear motion), the whole egg beater rotates when mixing eggs (rotational motion). Although in theory such objects can be manipulated by a robotic arm—by first grasping it using a gripper and then performing manipulation by a series of joint rotations to create the movement, there remain two limitations that prevent the robotic arm from manipulating such objects to perform tasks as well as humans: Speed and Range of motion.

204 2.1.1 *Speed limitation.* A robotic arm mostly manipulates objects at its end-effector, which generates large loads on the
205 arm due to the moment of inertia. Therefore, tasks such as knife chopping (Figure 4A1), which require a fast periodic
206 motion at the end-effector may exceed the robotic arm's capability.

Objects manipulated as a whole		Objects manipulated by constituent parts			
Linear	Rotational	Squeezing & Releasing		Twisting	Pumping
A1  Knife	B1  Screwdriver	One-directional	C1  spray bottle	D1  Door knob	E1  Hand sanitizer
A2  Spice can	B2  Whisk		C2  Wire cutter	D2  Pepper grinder	E2  Pump

→ Movement of the object when manipulated by human

Fig. 4. Everyday handheld objects are often not manipulable by a generic robotic arm with a common parallel gripper: when manipulating an object as a whole, speed and range of motion are two main limitations and; when manipulating an object by its constituent parts, the dexterity required to both grasp and perform a range of manipulation (squeezing/releasing, twisting and pumping) is the main challenge.

2.1.2 *Range of motion limitation.* Using a conventional robotic gripper, some object manipulation requires a certain type of grasp, e.g., rotating a screwdriver (Figure 3a). However, when the object is further way, even though it is still within reach of the robotic arm, the grasp is different and as such, it is no longer able to afford the same manipulation (Figure 3b). In other words, grasping the tool at a certain angle might render the manipulation impossible because reaching that angle would already constrain the robotic arm's joint rotation, thus limiting how it can perform subsequent manipulation (Figure 3b). In some edge cases, the manipulation might be interfered with by the physical surroundings, e.g., the ground (Figure 3c). In robotic terms, under such circumstances, the robotic arm is said to be outside of its *dexterous space* [34] when having to perform the manipulation at certain angles.

2.2 Objects Manipulated by Constituent Parts (Dynamic Objects)

In contrast to objects manipulated as a whole, there exist objects made up of and manipulated by their movable parts, as shown in Figure 4C-E. Manipulating such objects tends to be more difficult for a robotic arm, which needs to first grasp the object stably and then perform a dexterous manipulation like human hand(s), e.g., grasping a pepper grinder and then twisting the grinding cap (Figure 4D2). We summarize the following three types of manipulation that makes objects not manipulable by generic robotic arms, each of which can be either one- or bi-directional.

2.2.1 *Squeezing.* In order to manipulate objects that require ‘squeezing’ (e.g., a wire cutter in Figure 4C2), a robotic arm (e.g., with a common parallel gripper) would have to pick two points on the squeezing handles for a firm grasp, and then apply a steady force to squeeze the object. However, as the best grasping points for performing the squeezing manipulation (e.g., near the tip of the handle) are usually the furthest away from the center of gravity, this makes the grasp unstable and slippery.

261 2.2.2 *Twisting*. The twisting force applied to objects *e.g.*, a door knob (Figure 4D1) would produce a rotational torque
 262 on the robotic arm itself after securing the grasp, which may cause the whole system to become unstable. This is also the
 263 most common failure in the *DARPA Robotic Challenge* [30].
 264

265 2.2.3 *Pumping*. Different from the above, after grasping an object, the robotic arm needs an additional contact surface to
 266 perform the pumping task (*e.g.*, to press the hand sanitizer in Figure 4E1), which makes it nearly impossible to perform
 267 by a common gripper with a parallel design.
 268

269 Finally, note that some objects (*e.g.*, pepper grinder, pump) would require bi-manual manipulation. In other words, a
 270 *single* robotic arm, even as dexterous as a human hand, would find it challenging to manipulate these objects.
 271

272 3 RELATED WORK

273 3.1 Augmenting Generic Interfaces and Everyday Objects

274 Roman can be thought of as enabling two types of augmentation: (*i*) augmenting a generic robotic arm so that it can attach
 275 to and manipulate various handheld objects and (*ii*) augmenting everyday objects so that they become manipulable by the
 276 robotic arm. As such, we first review prior work on these two aspects.
 277

278 3.1.1 *Augmenting generic physical interfaces*. Existing work has explored various designs for extending the functionality
 279 of generic interfaces. For example, HERMITS extended the capability of self-propelled tangible user interfaces (TUIs) by
 280 designing mechanical shell add-ons for TUIs to dock and drive dynamic interactions [39]. Katakura *et al.* introduced
 281 a novel way of augmenting a 3D printer head into a robotic manipulator with the mechanical attachments printed by
 282 the printer itself [24, 25]. Other researchers augmented a pin-based display by the conversion of the mechanical motion
 283 of pins with passive mechanisms to enrich the pins' dynamic interaction [40, 49]. Davidoff *et al.* designed mechanical
 284 actuators based on the Lego MindStorm toolkit to operate on switches that were intended for humans only [12]. In
 285 comparison, Roman's mechanisms augment the physical interface of a robotic arm, *i.e.*, the end-effector, with a versatile
 286 magnetic gripper with a focus on assisting the robotic arm to manipulate different everyday objects that would otherwise
 287 be difficult to manipulate.
 288

289 3.1.2 *Augmenting everyday objects*. Another focus of previous research is extending the capability of everyday objects.
 290 Reprise makes handheld objects easy-to-manipulate by people with disabilities by generating adaptions attached to the
 291 object [8]. Medley enhanced the material properties of 3D printed objects by embedding different reusable materials into
 292 the model [7]. Romeo extended the default functionality of everyday objects by embedding a transformable robotic arm
 293 into the 3D model of the object [32]. RetroFab augmented the interactivity of physical interface by adding an enclosure
 294 consisting of mechanical and electrical components that could automate physical controls [46], which was later extended
 295 by Robiot in automating dynamic everyday objects, *e.g.*, adjusting a lamp's joint angle [33].
 296

297 RetroFab and Robiot employed *active* mechanism (*i.e.*, with motors onboard), which inevitably makes the augmented
 298 object bulky and expensive to scale to the numerous everyday objects in the environment. In contrast, Roman utilized a
 299 passively actuatable mechanism attached to an object with only mechanical components to be manipulated by a generic
 300 robotic arm.
 301

302 Since Roman proposes a different complementary approach for robotic arms to grasp and manipulate everyday objects,
 303 below we will review related work in robotics research, focusing on gripper design and manipulation.
 304

313 3.2 Relationship to Robotic Research

314 3.2.1 *Gripper design.* In Robotics, grippers are the most common type of end-effector and an important medium for
315 robots to interact with the real world. The design of robotic grippers has been extensively studied in academia and industry
316 by researchers and practitioners to explore designs of different types of robotic end effectors, *e.g.*, grippers for pick
317 and place, tight grasping, and more. There are two major strategies in designing a gripper: for general purposes or for
318 a specific task. Researchers have explored different types of general purpose robotic grippers including linkage-based
319 parallel mechanism robotic grippers [22, 28] and compliant underactuated robotic grippers [10, 31, 35, 38]. Meanwhile,
320 exploring how to configure task-specific grippers is an emergent topic that has gained recent attention. Feix *et al.* provided
321 a taxonomy to categorize the potential tasks and the corresponding design features of the robotic gripper [16]. Researchers
322 have designed grippers for different shapes of target objects [41], *e.g.*, for picking objects lying on flat surfaces [2, 4], for
323 assembly tasks [54], or for operating a heavy load [50].

324 Furthermore, the design of robotic grippers should not only consider the geometry of the objects, but also the interaction
325 with the environment and the kinetostatic properties of the grippers [3]. Different factors are considered in previous
326 research such as dynamic loads [42] and active surfaces after grasping [19, 43]. These are closely related to the tasks
327 conducted by robotic grippers in daily life interacting with everyday tools, which are summarized by the robotic grasping
328 and manipulation challenge [15]. Roman uniquely combines both strategies by implementing a general purpose versatile
329 gripper that could grasp different objects and object-specific mechanisms that could be driven by the gripper to achieve a
330 dynamic manipulation of everyday handheld objects.

331 Another important gripper-related topic is to create ‘tool changers’ that adapt to the different shapes and sizes of
332 everyday objects. There are different types of tool changers for a robotic arm: automatic tool changers that are electro-
333 mechanically actuated [20] or passive mechanisms actuated by a host robot [5, 45]. Similar to our method, researchers
334 also proposed the design of mechanical tool for robots to grasp different daily objects with modular two-finger gripper
335 but only focused on the grasping of objects with different sizes [23]. Adding to that, our method could solve a more
336 complex manipulation problem of manipulating tools with movable parts (Figure 4), which is almost unattainable using a
337 two-finger type end effector. Furthermore, since Roman provides customized mechanisms for different objects that can be
338 driven by the same custom gripper, our method provides a less complex method to program the motion of the arm than
339 solutions that incorporate tool changers.

340 3.2.2 *Manipulation.* Besides innovating the gripper design, researchers have also investigated methods of control and
341 task planning that program a given robotic arm to perform tasks in human environment. Some research focused on
342 developing a control strategy based on perception such as analyzing the geometry of the objects to obtain the best grasping
343 point [37, 52] or imitating human operation to open medicine bottles [13, 36]. Some developed new algorithms and system
344 designs for specific contexts such as grasping flat objects [47]. At the application level, robots can assist people with daily
345 living tasks which range from fetching a mug [17] to taking medications [29]. In contrast to the above approaches of
346 enhancing perception and control, Roman aims for a different and complementary goal of achieving manipulation by
347 augmenting everyday objects to be more manipulable by a generic robotic arm.

348 Different from traditional robotic research, a concept of *dexterous manipulation* was first defined by Okamura *et al.*
349 in which a robotic gripper moves objects from one configuration to another [44], *e.g.*, adjusting the angle of a phone in
350 the hand. Such a concept is still being actively studied by researchers in scenarios such as robotic in-hand manipulation
351 [1, 51, 55]. However, such manipulation requires precise control of the forces and motions of fingered or specialized
352 robotic hands and therefore cannot be accomplished by conventional robotic grippers. Roman, as a complementary
353

365 solution to dexterous manipulation, enables a conventional robotic arm to connect with a versatile magnetic gripper to
 366 grasp and manipulate everyday objects with dexterity enabled by object-specific mechanisms.
 367

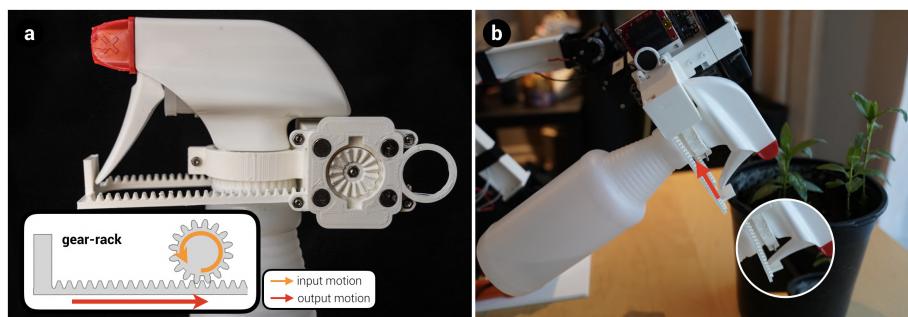
368 4 EXAMPLES: ROMAN MAKES HANDHELD OBJECTS ROBOTICALLY MANIPULABLE

370 We showcase a series of examples where Roman mechanisms attached to handheld objects make them more manipulable
 371 by a consumer-grade 6-DoF robotic arm (equipped with a Roman gripper). There are two major types of application
 372 scenarios where the objects need to be robotically manipulable: (i) A human collaborates with a Roman-equipped robotic
 373 arm, *e.g.*, the human would hand an object over to the robotic arm where it would then be picked up and manipulated, in
 374 tasks that the robot is expected to manipulate a lot of different handheld objects, *e.g.*, making a scrambled egg (Figure 6de,
 375 Figure 14, etc) and (ii) a Roman-equipped robotic arm takes the place of human in performing repetitive tasks by
 376 manipulating a hard-to-manipulate object, *e.g.*, a spray bottle. We focus on demonstrating the wide capabilities of Roman
 377 in enabling object manipulation, while discussing the technical details of the gripper design and the mechanism generation
 378 method later in subsequent sections.
 379

380 4.1 Manipulating Objects as a Whole

381 4.1.1 *Increasing speed.* As mentioned above, some objects may require a high speed of manipulation that exceeds the
 382 motor's capability at the end-effector. Figure 5 and Figure 6 showcase several different examples where Roman enables
 383 a high speed of manipulation: by attaching a pin-in-slot mechanism, the Roman gripper can drive a knife to perform
 384 repetitive high-speed vertical motion, *e.g.*, for chopping scallions (Figure 5ab), or to mimic a human's tapping motion on
 385 the spice bottle to spread enough white pepper (Figure 5cd). With a spur-gear mechanism, the Roman gripper enables a
 386 high speed of rotation of the whisk to mix an egg (Figure 6de).

387 4.1.2 *Expanding range of motion.* Roman helps expand the range of motion while manipulating specific objects. For
 388 example, grasping a screwdriver at certain distances/angles may prevent a conventional robotic arm from performing the
 389 rotating manipulation (Figure 3). Therefore, Roman adopts a bevel gear mechanism that could change the rotational axis
 390 of the input, expanding the range of motion of the screwdriver's manipulation (Figure 6a-c). Similarly, by attaching the
 391 whisk at a small angle (as opposed to being perpendicular) when fastened to the output spur gear, we can expand the
 392 range of rotational motion at its end (Figure 6de).



400
 401 Fig. 7. Configuration of the gear rack mechanism to produce linear motion (a) in a single direction to squeeze the handle of a spray
 402 bottle (b).
 403

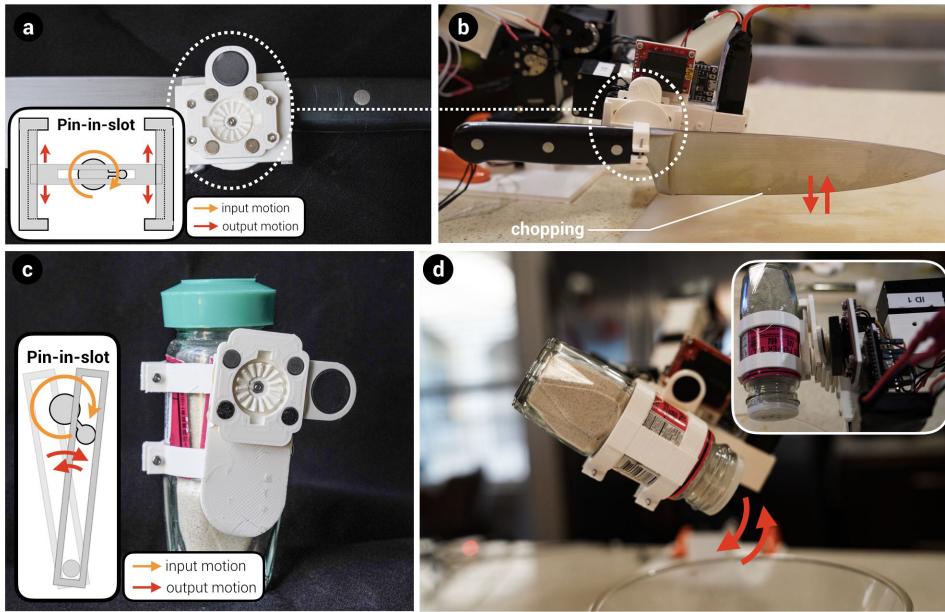


Fig. 5. Configuration of the pin-in-slot mechanism (a) to produce periodic up and down motion with a knife (b) and an alternative configuration of the pin-in-slot mechanism (c) to produce periodic side to side motion for spreading peppers from a spice bottle (d).

4.2 Manipulating Objects by Constituent Parts

4.2.1 *Enabling squeeze & release manipulation.* As shown in Figure 1, the wire cutter augmented with a gear-rack mechanism can perform the task of cutting a wire by squeezing the handles (Figure 1h). Similarly, a gear-rack is used for the spray bottle and thus a robotic arm could fetch the bottle and water the flower automatically (Figure 7). As the spray bottle only requires one-directional manipulation, the mechanism only needs to squeeze in one direction before releasing the handle to return to its original position. The chopsticks require a bi-directional squeezing and releasing manipulation to pick and place food and a spur gear mechanism is used for the manipulation (Figure 8). A gear-rack mechanism is also utilized for the can opener in order to perform the squeezing and rotating manipulation to pierce the peanut butter can (Figure 9dg). Specifically, a ratchet design (Figure 9ef) is used to ‘lock’ the squeezing of the handles (Figure 9h), which then enables the gripper to detach from the handles, attach to the cutting part (Figure 9c), and drive the rotation of the handle to open the can (Figure 9h).

4.2.2 *Enabling twist manipulation.* Following the manipulation of piercing the can, the robotic arm is able to twist the handle continuously to open the can by using a spur gear mechanism (Figure 9ch). Since both the squeezing and the twisting manipulation of the can opener require a relatively large strength to manipulate, a gear box with ratio of 9:1 is adopted to increase the output torque applied to the target object (Figure 19d). Using a pepper grinder is important in a series of cooking tasks (*e.g.*, making an omelette). With a bevel gear mechanism, the robotic arm can twist the grinder repetitively to sprinkle pepper on the eggs (Figure 11cd). A robotic arm can also collaborate with a human in a cooking task by opening the lid of a starch jar with a bevel gear mechanism (Figure 11ab). Roman also enables a robotic arm to open the door by twisting the door knob with a spur gear mechanism on it (Figure 10).

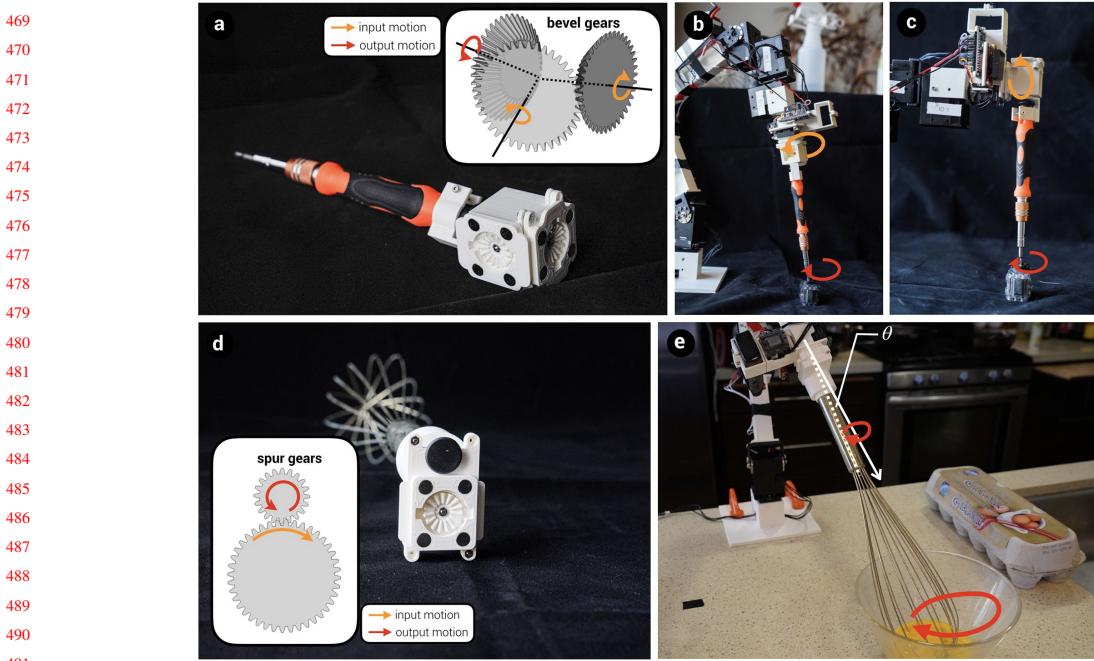


Fig. 6. Configuration of the bevel gear mechanism to produce rotational motion at an angle with a screwdriver (a), allowing for two attachment configurations (b, c); the spur gear mechanism with a reverse reduction gear ratio to produce higher rotational speeds with a whisk (d) and tilting the whisk by a small angle further expands the range of motion at its end (e).

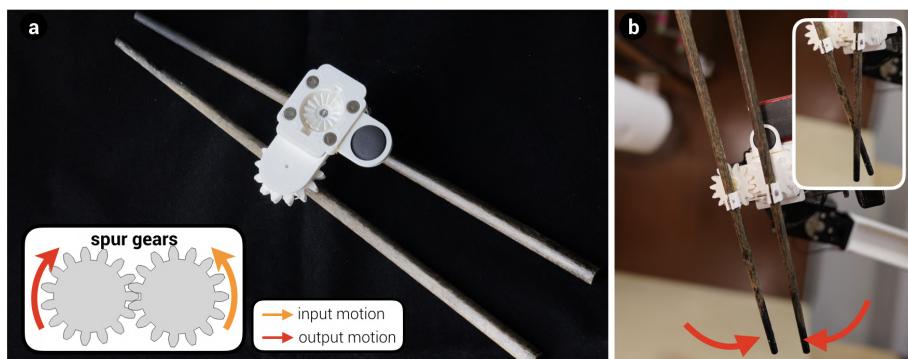


Fig. 8. Configuration of the spur gear mechanism to produce rotational motion to squeeze together the tips of chopsticks (a), and a practical demonstration of the chopsticks being manipulated with the mechanism attached (b).

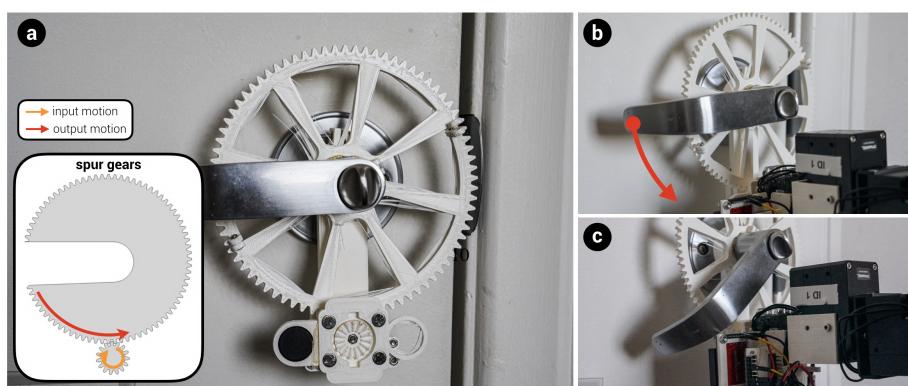


Fig. 10. Configuration of the spur gear mechanism with a reduction gear ratio to produce rotational motion with high torque in order to rotate the door knob (a), and a practical demonstration of the door knob being twisted with the mechanism attached (b, c).

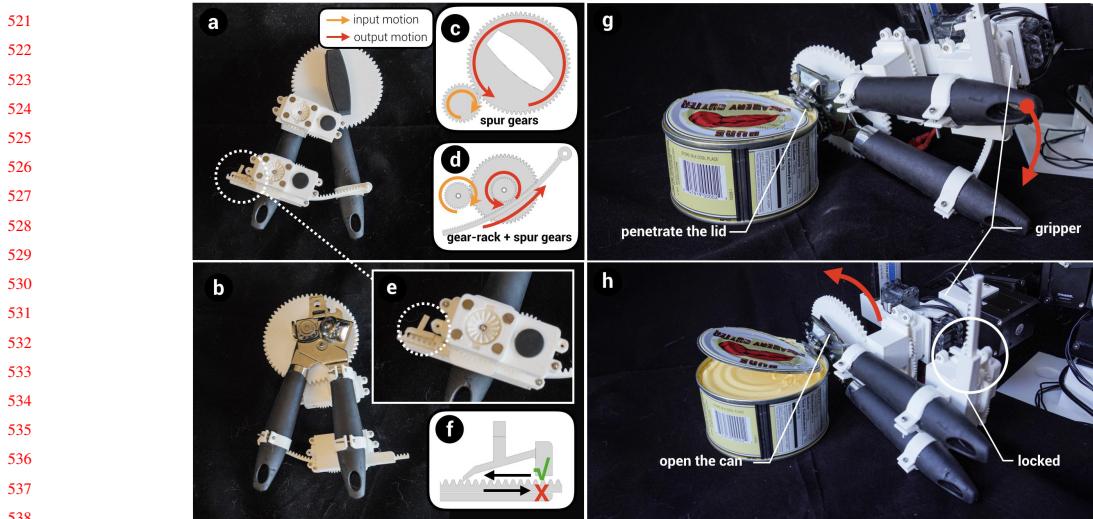


Fig. 9. Combination of two separate mechanisms with a reduction gear ratio to produce high torque: a spur gear to cut around the can (c), and a gear rack to pierce the can (also using spur gears to increase torque) (d). A ratchet mechanism (f) is used to maintain the position of the piercing mechanism (*i.e.*, keep the handles squeezed), which allows the gripper to switch to the other mechanism (c) for cutting the can open.

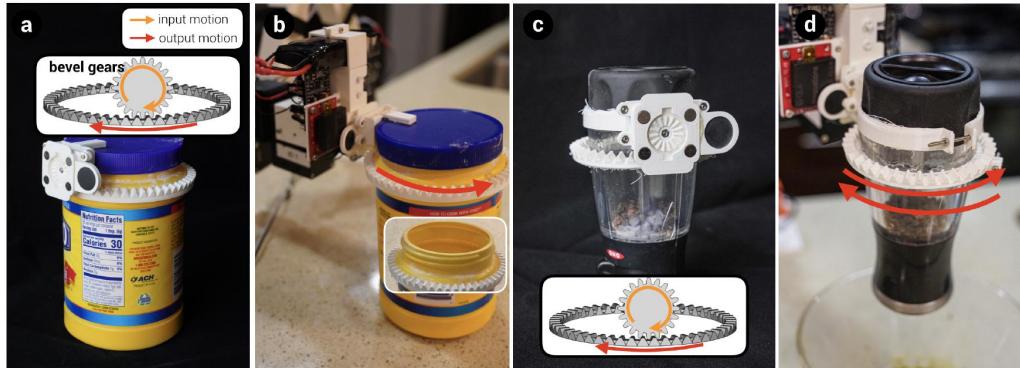
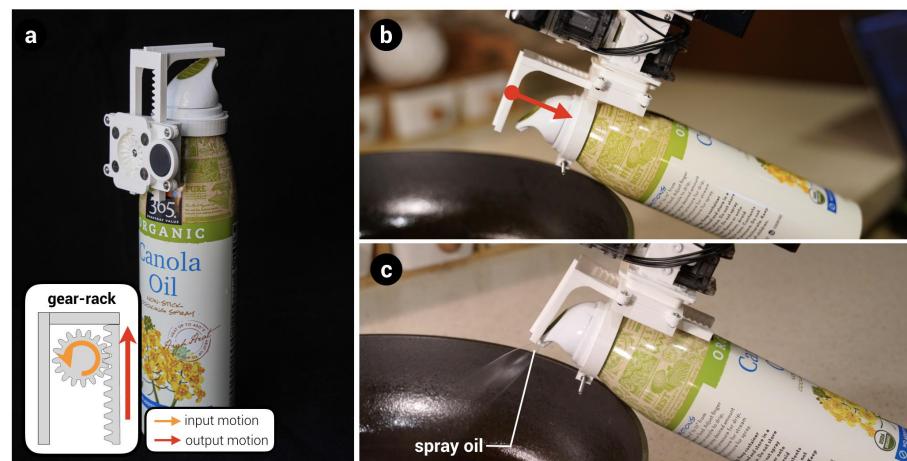
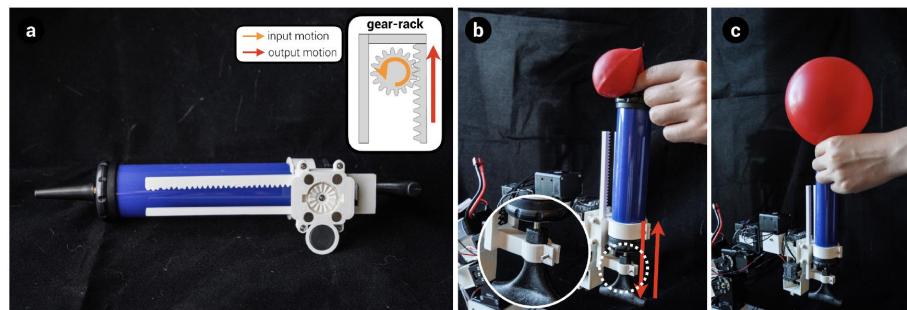
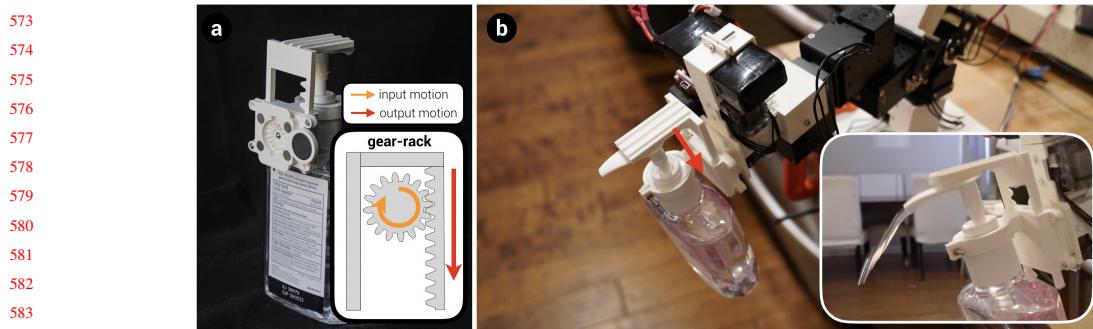


Fig. 11. Two examples of the bevel gear mechanism to produce rotational motion at an angle to unscrew the lid (a, b) or to rotate the pepper grinder (c, d).

4.2.3 Enabling pump manipulation. With a gear-rack mechanism, an oil spray can be manipulated by a robotic arm to help human cook (Figure 14). A robotic arm can fetch a bottle of hand sanitizer augmented with a gear-rack mechanism and pump it when the user approaches it (Figure 12). To enable a repetitive manipulation of the balloon pump, a gear-rack mechanism is also adopted to perform a bi-directional manipulation (Figure 13).



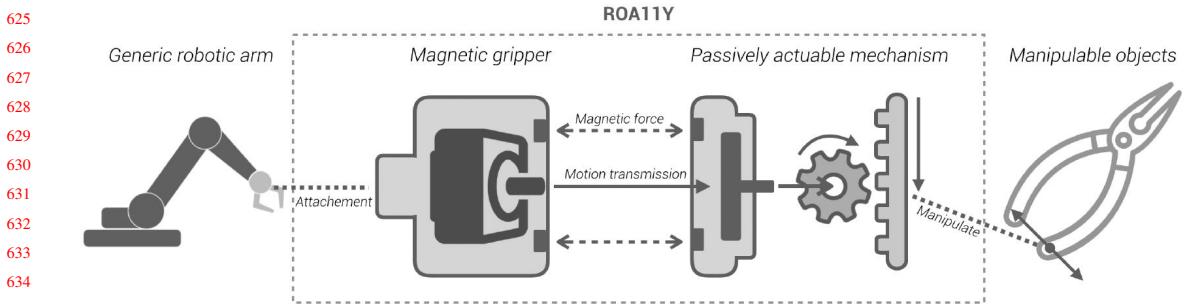


Fig. 15. Overall hardware structure of Roman

5 SYSTEM OVERVIEW OF ROMAN

Roman is an all-in-one solution to make everyday objects manipulable by generic robotic arms and includes both hardware and software support:

- **Hardware modules** consist of (as shown in Figure 15)
 - §6.1: A modular magnetic gripper that can attach to or detach from an object's add-on mechanism, recognize the object to retrieve the corresponding control program, and transfer the driving force from the robotic arm's motor to the mechanism to execute the object-specific manipulation;
 - §6.2: 3D-printable powerless mechanisms (spur gear, bevel gear, gear-rack, and pin-in-slot) attached to the object which enables objects to be manipulated as a whole or by their constituent parts. The mechanisms are easy to remove/assemble using screws.
- §7: **Software module** is a tool for robotic engineers to interactively specify custom motion profiles for manipulating a specific object (*e.g.*, amplitudes of a signal over time to be sent to the motor that drives the mechanisms to squeeze a wire cutter). We specifically focus on authoring motion profiles, which, to the best of our knowledge, is unsupported by prior work; meanwhile, the task of generating the 3D models of mechanisms—based on the type of motion and an object' geometry—can be supported by existing tools [6, 8, 32, 33].

6 HARDWARE IMPLEMENTATION

6.1 A Magnetic Gripper to Attach to, Recognize, and Transfer Motion to an Object's Add-on Mechanism

The magnetic gripper serves as the intermediary between the robotic arm with two main functionalities: 1) attaching to and detaching from the passively actuatable mechanism on the target objects and 2) driving the mechanisms on the target objects to perform the manipulation. Further, the gripper also contains an RFID reader for recognizing which object it is attached to and running the corresponding control program.

6.1.1 *Attaching and detaching mechanisms:* The gripper uses four neodymium magnets (Figure 16D) to generate the magnetic force for attaching to the mechanism on the target objects. The four neodymium magnets could generate a pull force (the vertical force required to pull the magnets from a steel surface) of 11.2lbs in total, which is sufficient to securely attach to common everyday handheld objects. Furthermore, the strong magnetic force enables the gripper to attach to objects when it is within approximately 1cm of them, which increases the fault tolerance of the robotic arm's

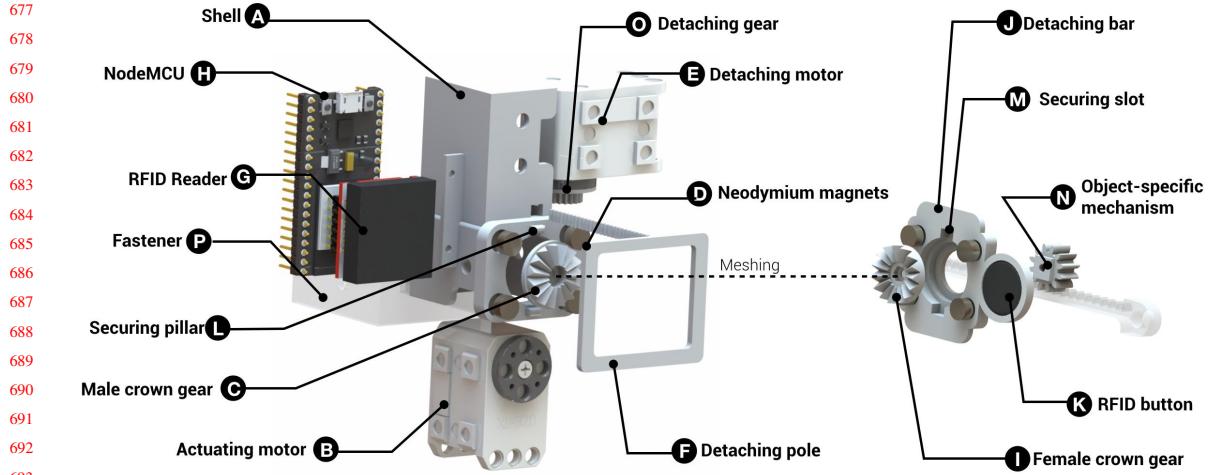


Fig. 16. Exploded view of the Roman gripper.

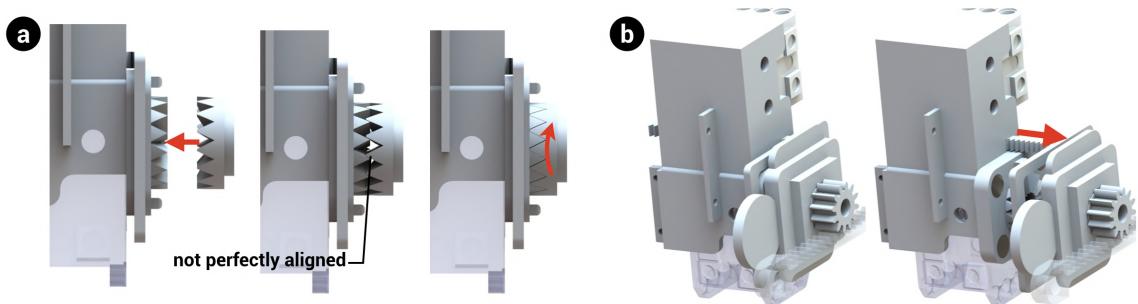


Fig. 17. Meshing (a) and detaching (b) operations of Roman gripper.

manipulation (e.g., a low cost robotic arm³ may have accuracy larger than 5mm). To further strengthen the connection, we designed a pin structure (Figure 16L) to counter the lateral force generated during the actuation of the mechanisms.

On the other hand, a stronger connection means larger force required for the detachment. To provide auto-detachment of the mechanisms, Roman employs a Dynamixel XL-320 motor⁴ (Figure 16E) and a gear-rack mechanism (Figure 16F/O), which could transfer the rotational motion of the motor into linear motion, for the auto-detachment (Figure 17b). While the XL-320 motor can generate a maximal torque of 0.39 N·M, we designed the gear of the gear-rack mechanism to have a radius of 5mm which enables it to generate 17.52lbs of force on the rack for detachment (Figure 16O).

6.1.2 Motion transmission: The gripper also serves as the actuator of the mechanisms on target objects. Same as the detachment, Roman employs an XL-320 motor (Figure 16B) and a pair of crown gears (Figure 16C/I). The crown gear could transmit the rotational motion of the motor to the mechanism with auto-alignment of the teeth (Figure 17a). On the

³PincherX 150 Robotic Arm: <https://www.trossenrobotics.com/pincherx-150-robot-arm.aspx>

⁴<https://www.robotis.us/dynamixel-xl-320/>

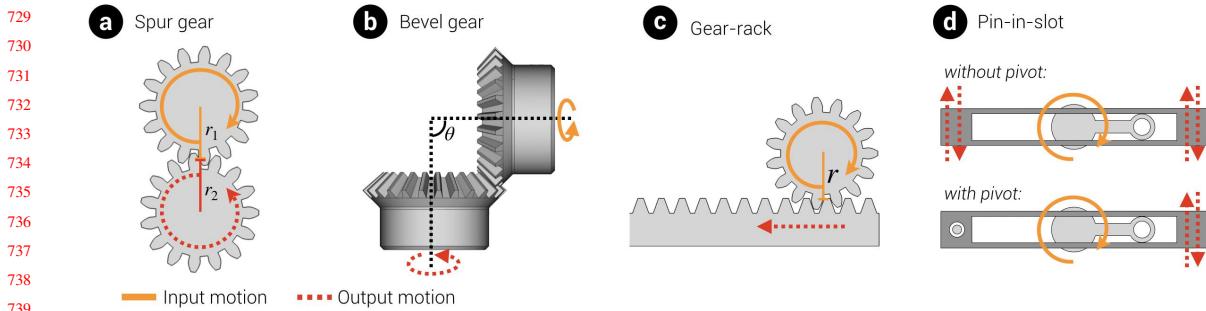


Fig. 18. Overview of the four types of mechanisms used to achieve target motions: spur gears for rotational motion (a), bevel gears for rotational motion at an angle (b), gear rack for linear motion (c), and a pin-in-slot mechanism to achieve periodic motion (d).

side of the mechanism, modular driving gears (e.g., gears or gear-rack mechanism, Figure 16N) can be assembled with the female crown gear to actuate different types of mechanisms.

6.1.3 Communicating with target object. Roman employs NodeMCU ESP-12 module⁵ for communicating with the web server (Figure 16H). An RFID reader is used for recognizing different objects (Figure 16G) and each mechanism on the object comes with an RFID tag (Figure 16K) whose ID is associated with a user-defined control program to perform a specific task. Fasteners can also be customized and 3D printed to fit different robotic arms using screws (Figure 16P). Finally, the gripper can be powered by an additional 7.4v LiPo battery. The whole system weighs 110g without the battery, which makes it possible to be installed and used on any generic robotic arm.

6.2 Mechanisms for Manipulating Handheld Objects with Different Motion Profiles

We first discuss the mechanism design on the objects' side that transfers the rotary input from the motor into task-specific motion profile for different objects.

Objects require different motion profiles in order to perform their tasks. The motion profile is defined as the required output motion for the objects in order to perform an object-specific task. For example, objects in the *squeezing* category may require a curved motion profile (e.g., the legs of the wire cutter move in an arc trajectory for a cutting task Figure 1); objects in the *twisting* category require a rotary motion profile (e.g., rotating the door knob to open the door Figure 10) and objects in the *pumping* category require a linear motion profile (e.g., linear pushing motion for the balloon pump Fig Figure 13).

To address this, Roman adopts four basic types of mechanical mechanisms: spur gears, bevel gears, gear-and-rack and pin-in-slot mechanisms (Figure 18). These four types of mechanisms are able to transfer the rotary motion from the motor into different motion profiles.

6.2.1 Spur gears. Spur gears are a mechanism in which multiple gears mesh together to transmit the rotary motion from one shaft to another (Figure 18a). Therefore this mechanism can transfer the rotary motion from the motor input into a rotary motion profile.

Rotary to rotary While the output motion has the same rotary motion as the input, the gear-pair mechanism can translate the rotary axis to a parallel position, which enables the mechanism to be anchored to a fixed point on the object

⁵<https://www.nodemcu.com/>

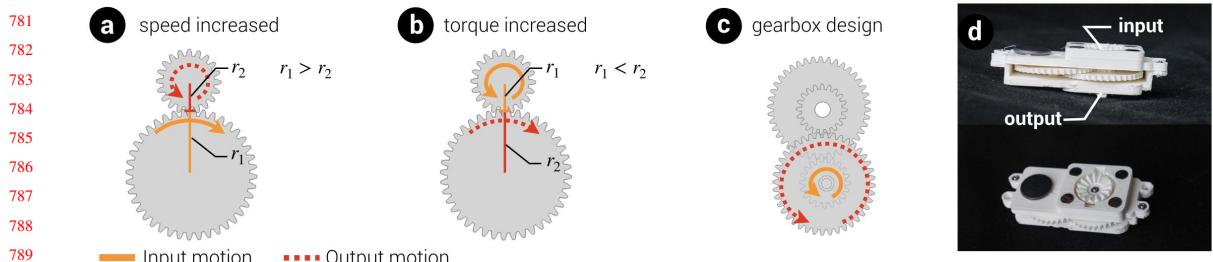


Fig. 19. Breakdown of the configuration of the gearbox, showing how to produce higher speeds at the expense of torque (a) and to produce higher torque at the expense of speed (b), as well as the physical configuration of the gears (c) and the design of the gearbox (d).

while manipulating the object. As shown in Figure 10, instead of being directly anchored to the door knob, which may get in the way of people using it, the mechanism for driving the knob can be shifted to a position where it does not interfere with regular use of the knob.

Speed and strength Besides the translation of the rotational axis, Roman leverages the property of reduction in gears to generate higher speed or strength than is typically available from a motor. Gear reduction is an arrangement of gears in which an input speed can be raised at the expense of torque, or the output torque can be raised at the expense of speed as explained in the following equations:

$$\text{Speed}_{\text{output}} = \frac{r_{\text{input}}}{r_{\text{output}}} \times \text{Speed}_{\text{input}}$$

$$\text{Torque}_{\text{output}} = \frac{r_{\text{output}}}{r_{\text{input}}} \times \text{Torque}_{\text{input}}$$

With a proper selection of the gears, the mechanism can generate much larger torque than the motor's base ability (Figure 19). In the meantime, with a reversed reduction gear, the mechanism can generate higher rotational speed than the motor's typical maximum speed. The whisk is an example of leveraging reversed reduction gears to enable the whisk to rotate at a high speed to beat the egg (Figure 6de). Such reduction gears can also be combined with other mechanisms depending on the task (*e.g.*, the can-opener adopts a reduction gear box over its gear-rack mechanism to generate enough force to pierce the can as shown in Figure 9d).

6.2.2 Bevel gears. Similar to the spur gears, the output of the bevel gears is also rotary motion. Unlike spur gears, bevel gears change the orientation of the rotational axis, which enables robotic arms to execute a twisting task from different angles (Figure 18b).

Rotary to rotary Bevel gears can generate a rotary motion profile for target objects. With the property of being able to change the orientation of the rotational axis, bevel gears enable robotic arms to operate twisting tasks with a space-efficient solution. For example, in order to perform a lid opening task, a spur gears design would make the mechanisms bulky as the driving gear will protrude from the jar lid. In contrast, A bevel gears design rotates the protruding gear so that it reduces the overall volume occupied by the entire mechanism (Figure 11).

6.2.3 Gear-rack. The gear-rack mechanisms are utilized to convert the rotary motion from the gear into the linear motion of the rack (Figure 18c). As the gear-rack mechanism generates a linear motion profile, this mechanism can be used by the objects in the *grasp & pump* category. Further, the gear-rack mechanism can also be utilized for generating a curved motion profile, which will be discussed below.

833 Rotary to linear While the gear-rack mechanism converts the rotary motion from the gear into the linear motion of the
834 rack, it could help the robotic arm to manipulate objects that require pumping (Figure 4E). Designers of the mechanism
835 could adjust the output velocity and force by modifying the size of the gear based on the equation:
836

$$\text{Speed}_{\text{output}} = \text{rpm}_{\text{motor}} \times r$$

$$\text{Force}_{\text{output}} = \frac{\text{torque}_{\text{motor}}}{r}$$

841 where r represents the radius of the gear attached to the motor. For example, hand sanitizer is an example that requires
842 large torque to squeeze out sanitizer Figure 12.

843 Rotary to curved Besides the linear motion profile, the gear-rack is also able to generate a curved motion profile for
844 squeezing objects such as the squeezing leg of can opener (Figure 9). Roman leverages the compliant property of the
845 PLA material to design racks that can bend themselves to adapt to the curved motion (Figure 9e). As a result, the fastener
846 on the rack (normally attached with the other movable part of the object) would require an active joint in order for the
847 rack to rotate relatively, without which the curved motion may generate a large offset at the tip of the rack and break the
848 mechanism.
849

850 Bi-directional manipulation The gear-rack mechanism may deal with objects requiring either one or bi-directional
851 manipulation. For objects with only one-directional manipulation such as the oil spray (Figure 14), a simple bar is
852 sufficient for generating a one-directional force. However, objects that require bi-directional manipulation such as the
853 balloon pump, require an additional structure fastened to the part to enable motion in two directions (Figure 13b).
854

855 Single-direction constraints Roman also provides a single direction constraint using the gear-rack mechanism. For
856 example, in order to perform the task of piercing the can and then opening it, the mechanism needs to lock the squeezing
857 mechanism for the opener to continue to pierce through the surface of the can. To achieve this, we employ a design
858 inspired by the ratchet gear, where the rack can move freely in one direction while being prevented from moving in the
859 opposite direction unless a pin is depressed, which releases the mechanism (Figure 9d).
860

861
862
863
864
865
866 6.2.4 *Pin-in-slot*. A pin-in-slot is a mechanism where a pin-joint moves along or parallel to a slide-joint (Figure 18d).
867 The pin-joint receive the rotary input from the motor and transfer the motion to the slide-joint. By fixing one side of the
868 slide-joint, the pin-in-slot mechanism can generate a single-sided motion profile (Figure 18d bottom) while a double-sided
869 motion profile can also be generated by placing the slide-joint inside of another slider joint (Figure 18d top).
870

871 The spice bottle is an example of using the pin-in-slot mechanism to mimic the human shaking action by generating a
872 single-sided motion profile (Figure 5cd) and the knife is an example of performing the chopping task by generating a
873 double-sided motion profile (Figure 5ab).
874

875 Periodic motion Besides generating linear and curved motion profiles, the pin-in-slot mechanism is good at generating
876 a periodic motion. As shown in Figure 20, the continuous rotation of the pin-joint can generate a double-sided or single-
877 sided periodic motion at the slide-joint, e.g., for continuously shaking a spice bottle (Figure 5cd). While it is possible
878 for other mechanisms to generate a periodic motion with periodically changed control input, the control signal may
879 experience data loss when changing intensely in short periods. Differently, the periodic motion generated by the pin-in-slot
880 mechanism is easier to control because it relies on the stability of the mechanism components.
881

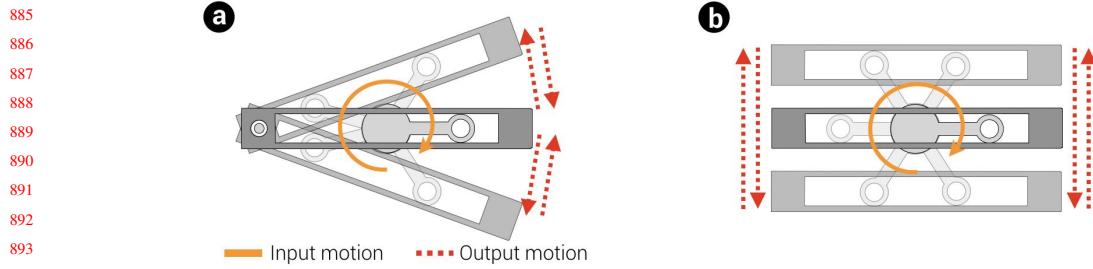


Fig. 20. Demonstration of the two types of periodic motion that could be accomplished with a pin-in-slot mechanism: angular (a), and up and down (b).

7 SOFTWARE IMPLEMENTATION: A TOOL FOR ROBOTIC ENGINEERS TO SPECIFY CUSTOM MOTION PROFILES TO MANIPULATE OBJECT-SPECIFIC MECHANISMS

In this section, we illustrate the workflow of a user authoring the control program for a task given some pre-generated mechanisms to manipulate a target object. We assume that the users are robotic engineers with a mechanical or robotic background and the mechanism generation is supported elsewhere by tools similar to [8, 21, 33]. We exclusively focus on the less-supported part where the users author control programs for different object-specific tasks. Roman's front end is written in JavaScript and the back end is written in Python. Roman communicates with the ESP8266 module on the NodeMCU and stores the motion profile through a Python-based web server. The RFID reader on the gripper reads the ID of different RFID chips on the objects and accesses the corresponding motion profile via the NodeMCU.

7.1 Custom Motion Profiles

As Roman transfers the rotary motion input from the motor to a customized output motion profile, it requires further specification of the motion profiles in order to perform an object-specific task. For example, to design a task for manipulating the wire cutter while collaborating with a human user (Figure 1), the wire cutter is expected to (*i*) move to the desired position for the user to hold the wire for cutting, (*ii*) squeeze the handles about half-way and hold the position in order for the user to align and adjust the wire, (*iii*) squeeze fully to cut the wire, and (*iv*) return the handles to the initial configuration. While step (*i*) is conducted by the movement of the robotic arm, the rest are done by Roman hardware and require a custom design of the motion profile. More specifically, the customizability of the motion profiles amounts to specifying the amplitudes u of the control signal over time, which corresponds to the rotational speed of the motor output. The aforementioned task for the wire cutter has a motion profile as the control signal shown in Figure 22.

7.2 User Interface

As shown in Figure 21, to facilitate the design of such custom motion profiles, the Roman software provides a user interface for interactively specifying the custom motion profiles by adding or adjusting the key points in the graph of a control signal (Figure 21c). The u for the y-axis is a unitless value ranged from -1 to 1 which represents the rotational speed of the motor relative to the maximum speed under the current load (with negative values corresponding to rotation in the opposite direction). The x-axis represents time and the user has the ability to modify the total time the motor is running at a certain speed by adjusting the length of each period of the control signal. With the combination of different control signals, the user could design a custom motion profile for a object-specific task.

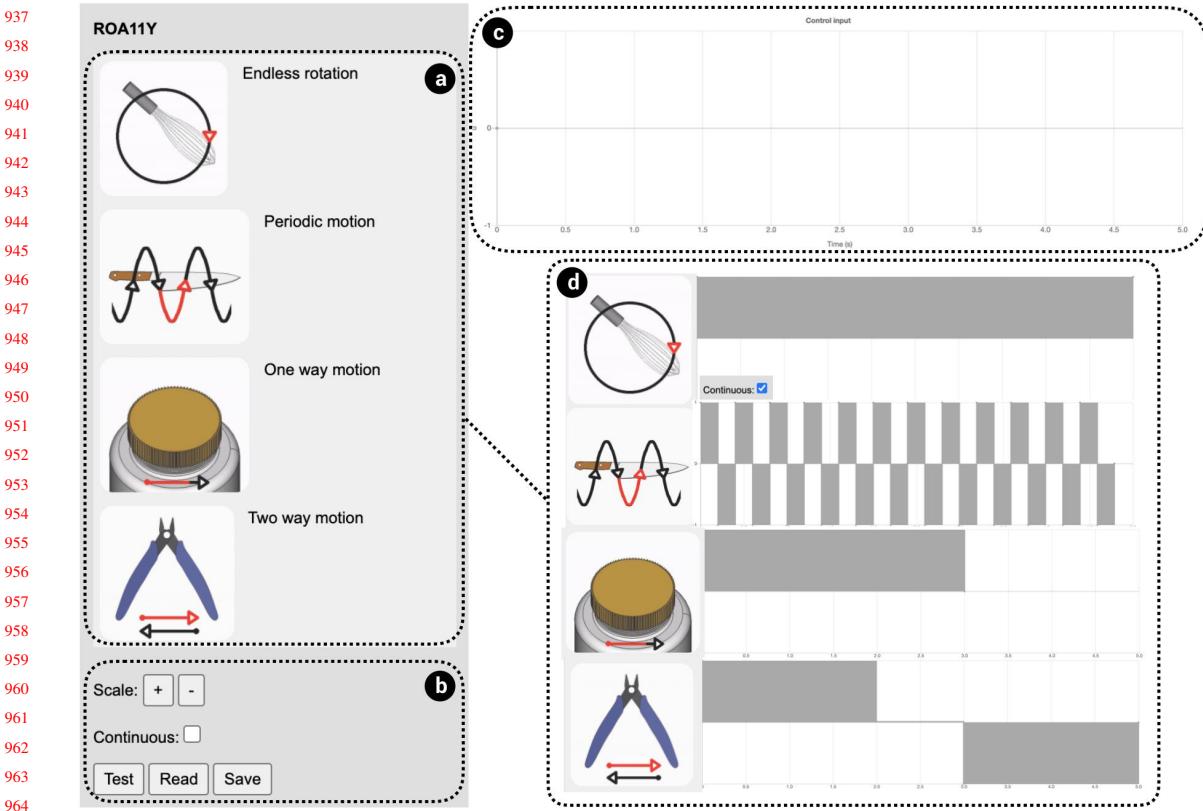


Fig. 21. The Roman user interface used for selecting motion templates, authoring control programs, and uploading them to the robotic arm.

7.2.1 *Motion Profile Templates and Interactive Graph Editing.* Roman provides different ways to interact with the graph of the control signal. As shown in Figure 21a and d, Roman provides four types of motion profile templates for a user to adapt to their needs, including endless rotation (e.g., whisk), periodic motion (e.g., knife), one-way motion (e.g., jar lid), and two-way motion (e.g., wire cutter). By clicking on the corresponding template, the control signal is imported into and visualized on the interactive graph and the user can further adjust the motion profiles.

To start editing the graph, a user simply double-clicks anywhere on the graph and then adjusts the position of an existing point by dragging it. The user can also adjust the range of the x-axis (*i.e.*, increase/decrease time) by clicking on the +/- signs (Figure 21b), which either adds or removes 1 second.

7.2.2 *Real Time Testing.* There is uncertainty in manipulating real-world objects, *e.g.*, the manipulation of a wire cutter (the motor's rotational speed) might need to vary with different wires. As such, Roman enables a real-time testing mode (Figure 21b) that creates a feedback loop in which the user could test and adjust the custom motion profile to see how well the manipulation is performed on the target object. The user could also check the box of 'continuous' to specify a repetitive motion. Once a user has tested and is satisfied with the motion profile, they can click on the 'save' button to store the motion profile to the target object.

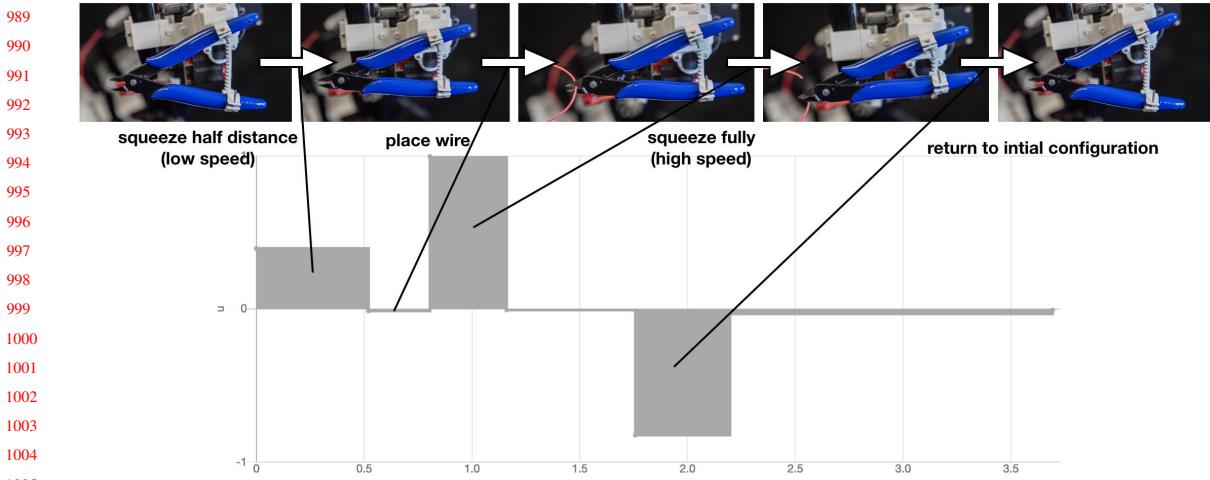


Fig. 22. A sample motion profile used to control the wire cutter in order to cut a wire.

8 VALIDATION WITH EXPERT INTERVIEW

We conducted an expert interview as an initial step to evaluate Roman. The purpose of this interview is to evaluate the usefulness and practicality of Roman in helping robotic engineers achieve object manipulation tasks, as well as to gather feedback and further suggestions from experts in the field of robotics.

8.1 Participants & Procedure

Four participants (all male with an average age of 28.25) with expert knowledge in the domain of robotics and mechanism design were invited for interviews. Three of them are Ph.D. students in the Mechanical Engineering Department with a focus on the research of robotic design and control (P1-P3). Among these, two had prior experience in developing robotic manipulators as a product in a robotic start-up (P1 and P2). The other participant is a post-doc researcher in Electrical Engineering Department with a Ph.D. degree in mechanical engineering and his research focuses on the computational design for origami robots (P4). Although no participants worked in the exact area of robotic manipulation in a human environment, all were familiar with the concept of mechanism design and control of robotic arm manipulation.

Each of the participants was interviewed in-person. First, they were explained the goal and function of Roman hardware and one author showed them demos of using the Roman gripper to manipulate five different objects (wire cutter, hand sanitizer, jar lid, whisk and spice can). These objects covered the four types of mechanisms we used in Roman as well as the four motion profile templates in the user interface. Afterwards, they were introduced to the Roman software and were asked to replicate the custom motion profiles of the wire cutter (which has the most complicated motion profile as shown in Figure 22). The motion profiles designed by the participants were test by performing a cutting task (similar to Figure 1) of an AWG 18 wire. The participants were able to design the profile and cut the provided wire physically in only 5 minutes (excluding the time needed to become familiar with interacting with the user interface) We then conducted a semi-structured interview, mainly to collect their feedback on the capability and practicality of Roman and potential directions for future research. The entire study lasted on average 25 minutes per participant.

1041 8.2 Results and Feedback

1042 All the participants reacted positively to Roman and gave feedback on both the usefulness and practicality of Roman.

1043
1044 *8.2.1 Roman was considered complementary to current approaches in robotic manipulation in human environment.* All
1045 the participants considered Roman to be a complement to the current research in the robotic manipulation in situations that
1046 require robotic arms in human environment. P2 stated that most of the existing robotic arms have limited functions or they
1047 required complicated programming of the control to use different types of tools. He thought that Roman provided a simple
1048 solution as the object + mechanism can be pre-designed as a product and delivered to customers. From the perspective of
1049 a user, P2 was initially concerned about the practically of possessing a robotic arm in daily life, which was later answered
1050 by himself when he noted that the Roman solution could make a robotic arm adapt to different objects in a task-heavy
1051 scenario like cooking. P1 was concerned about the pre-design of the custom motion profiles to be time-consuming, but he
1052 also pointed out that in the application scenarios such as cooking, the number of target objects is limited and therefore it
1053 is acceptable. P3 considered Roman usable, but he was more concerned about the necessity of the usage of robotic arm in
1054 such scenarios versus designing more easy-to-manipulate objects for human (*e.g.*, designing a jar that can be opened by
1055 pressing a button instead of adding mechanism to be manipulable by robotic arm). P4 mentioned that it makes more sense
1056 to have a strong and functional robotic hand to manipulate different objects without altering them but he agreed that the
1057 artificial hand solution is costly (a powerful enough robotic hand normally costs more than \$10k) and Roman could be a
1058 low-cost solution.

1059
1060 *8.2.2 Roman's mechanism designs and custom motion profiles were considered replicable by robotic experts.* When asked
1061 about whether they are able to replicate the hardware design of Roman's five examples⁶, all the participants agreed and
1062 were able to understand the rationale behind the design of each mechanism. P1 thought that it would be easy for a person
1063 with fundamental knowledge of mechanical design to replicate all of the examples. P2 also pointed out that with the
1064 explanation of different mechanisms as in the previous section (§6.1.3) as a guideline, it is a lot easier to select and design
1065 the mechanisms for specific objects. P4 stated that it is easy for robotic experts to replicate the examples but it would be
1066 great if a parametric design tool is provided for novice users.

1067
1068 All the participants were able to replicate the custom motion profiles for the wire cutter to perform a wire cutting task.
1069 All the participants could understand the differences between each motion profile template. P2 also valued the provided
1070 templates because, with the templates, he could know the overall movement of the mechanism and would only need to
1071 adjust a few parameters.

1072
1073 *8.2.3 Roman's hardware could be further improved.* The participants suggested that the hardware components can be
1074 improved to make the overall structure smaller and more durable including developing new types of mechanism (making
1075 the mechanisms more versatile), replacing the neodymium magnets with electromagnets (making the mechanisms smaller)
1076 and printing the custom mechanism in carbon (making the mechanisms stronger).

1077 8.3 Summary

1078 Results of the expert interview indicated that Roman could be a potential solution for robotic manipulation to assist human
1079 with daily tasks in certain scenarios. Also, it was considered practical for people with a robotic or mechanical engineering
1080 background to design and replicate the objects with mechanisms using Roman. The participants also recommended that a
1081 tool for novice users to custom such mechanism for existing objects would be valuable.

1082
1083 ⁶Note that participants did not actually replicate those examples, considering the time-consuming process of modeling, printing and assembly.

1093 9 LIMITATIONS, TRADE-OFFS, AND FUTURE WORK**1094 9.1 Performance testing**

1095 Roman's main contribution is the idea of adding mechanisms to enhance objects' manipulability, which complements
1096 existing approaches focused on gripper design and manipulation algorithms. As such, we chose to validate Roman by
1097 creating various examples to show the idea's wide applicability and by interviewing four robotic experts to obtain their
1098 initial feedbacks and reactions that would set the scene for future work. As next steps, future work that uses Roman on a
1099 specific type of objects/tasks (*e.g.*, manipulating a wire cutter) should define metrics for success of each task (*e.g.*, a wire
1100 is completely cut in one trial), control pertinent variables (*e.g.*, the thickness of wires), and employ repeated measures to
1101 obtain such metrics.

1102 9.2 Mechanisms interfering with normal use

1103 As Roman attaches 3D printed mechanism onto the existing object, some of the mechanisms may interfere with the
1104 normal use of the objects. For example, the spur gear mechanism on the chopsticks obstruct the way of a human would use
1105 them (Figure 8) and the bulky mechanism on the can opener also makes it hard for human to manipulate (Figure 9). While
1106 some of the mechanism can be easily disassembled, *e.g.*, the rack of the hand sanitizer can be easily pulled out (Figure 12),
1107 future work can focus on making the mechanisms modular and easy to disassemble such as LEGO MindStorm [27]. It is
1108 also possible to embed such function during the product design stage that aims at making the overall object + mechanism
1109 manipulable by both humans and robotic arms.

1110 9.3 The need to incorporate sensing & perception

1111 Roman focuses on enhancing manipulability by enabling human collaborating with Roman-equipped robotic arm or
1112 a Roman-equipped robotic arm to perform tasks independently. However, the latter scenario assumes that the object's
1113 position and orientation are known, which is a trade-off of not integrating sensing modules in the current design. Given
1114 the plethora of work on sensing and perception (*e.g.*, [48]), future work can add such modules to Roman's hardware
1115 components, which are expected to work independently and complementarily to the current set-up.

1116 9.4 Scale of the task

1117 Currently, Roman only focuses on tasks driven by the motor of the gripper while movements made by the other joints of
1118 the robotic arm have to be manually configured. As such, a Roman mechanism only affords manipulation in a limited
1119 space and cannot enable large-scale tasks, *e.g.*, holding a spatula to make stir-fry. Future work may extend Roman to
1120 include software support involving the entire robotic arm, *e.g.*, linking the action of the robotic arm with the actuation of
1121 the mechanism. With that, the Roman hardware could collaborate with the robotic arm to achieve more complicated tasks
1122 *e.g.*, scooping ice cream (enabled by Roman mechanisms) and distributing it into different locations (enabled by the rest
1123 of the robotic arm) automatically.

1124 9.5 Trade-off between torque and the size/complexity of mechanisms

1125 Currently, Roman enables the manipulation of objects by generating motions using a low-cost motor and 3D printed
1126 mechanisms. As a result, Roman is limited to supporting objects that do not require a lot of torque. While Roman provides
1127 a partial solution by adding a gearbox in between the gripper and the object to increase the torque (see the can opener
1128 example Figure 9), there is a trade-off between the size of the mechanism and the maximum torque the gripper could
1129 provide.

1145 generate. For example, manipulating a hedge trimmer (too large) or a sealed jar (too tight) will require rather bulky
1146 mechanisms unrealistic to be attached to the object. Substituting the motors in the current design with stronger models
1147 could partially solve the problem. Alternatively, there exist research opportunities to solve this problem with improved
1148 mechanisms, *e.g.*, using more durable materials such as metal linkages or cables to increase the generated force.
1149

1150
1151 **9.6 Manipulating objects with multiple movable parts or multiple consecutive manipulations**
1152 Some objects are articulated with multiple movable parts, *e.g.*, multi-functional pliers, Swiss army knife, flexible selfie
1153 rods, and Rubik's cubes. Technically, Roman's mechanisms can extend to more parts by enabling one additional motion at
1154 a time, yet, practically, too many mechanisms might not be fittingly added to an object and might even interfere with one
1155 another. Future work could focus on involving different materials (cable, metal, carbon) to make the mechanism smaller
1156 while functional.
1157

1158 Other objects might require consecutive manipulations to perform a task, *e.g.*, a corkscrew requires a twisting motion
1159 on the handle to penetrate a wine cork and a subsequent squeezing motion on the two arms, followed by pulling out the
1160 same handle. Currently Roman needs to provide separate mechanisms for each motion, which could result in the overall
1161 mechanisms too bulky. To address this problem, future work could incorporate interactive trajectory design into the
1162 mechanism design of Roman, similar to the approach in [11] that would need only one custom mechanism to accomplish
1163 multiple consecutive manipulations.
1164

1165
1166
1167 **9.7 Opportunities to support novice users**
1168 As mentioned by the robotic engineers in the interview, while it is easy for people with knowledge of robotics to replicate
1169 examples in Roman or design mechanism for new examples, novice users might struggle to design mechanism and
1170 understand which mechanism to use and how to specify the control signals since Roman does not provide 3D modelling
1171 support. One possible idea for future work is to instrument sensors on an object similar to the current Roman mechanisms.
1172 Then a user can demonstrate how they would manipulate an object, which can be captured by such sensors (*e.g.*, inertial
1173 measurement units or reflective markers + external optical tracking) and incorporate the algorithm in Robiot [33] which
1174 could generate the mechanism and control program automatically. In this way, Roman would be more usable to novice
1175 users.
1176

1177
1178 **9.8 Areas of hardware improvement**
1179 The hardware of Roman can be improved by: (*i*) switching to a half-duplex enabled microcontroller board that could
1180 enable the control of the mechanism to be self-adaptive to different loading conditions; (*ii*) replacing the neodymium
1181 magnets with electromagnets that can be programmatically controlled for attachment/detachment, thus dispensing with
1182 the additional motor currently used for detachment.
1183

1184
1185 **REFERENCES**
1186
1187 [1] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert,
1188 Glenn Powell, Alex Ray, et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research* 39, 1 (2020), 3–20.
1189 [2] Vincent Babin and Clément Gosselin. 2018. Picking, grasping, or scooping small objects lying on flat surfaces: A design approach. *The International
1190 Journal of Robotics Research* 37, 12 (2018), 1484–1499.
1191 [3] Vincent Babin and Clément Gosselin. 2020. Mechanisms for Robotic Grasping and Manipulation. *Annual Review of Control, Robotics, and
1192 Autonomous Systems* 4 (2020).
1193 [4] Vincent Babin, David St-Onge, and Clément Gosselin. 2019. Stable and repeatable grasping of flat objects on hard surfaces using passive and epicyclic
1194 mechanisms. *Robotics and Computer-Integrated Manufacturing* 55 (2019), 1–10.
1195

- 1197 [5] Ron Berenstein, Averell Wallach, Pelagie Elimbi Moudio, Peter Cuellar, and Ken Goldberg. 2018. An open-access passive modular tool changing
 1198 system for mobile manipulation robots. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 592–598.
- 1199 [6] Xiang'Anthony' Chen. 2016. Making Fabrication Real. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*.
 1200 17–20.
- 1201 [7] Xiang'Anthony' Chen, Stelian Coros, and Scott E Hudson. 2018. Medley: A library of embeddables to explore rich material properties for 3D printed
 1202 objects. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–12.
- 1203 [8] Xiang'Anthony' Chen, Jeeeon Kim, Jennifer Mankoff, Tovi Grossman, Stelian Coros, and Scott E Hudson. 2016. Reprise: A design tool for specifying,
 1204 generating, and customizing 3D printable adaptations on everyday objects. In *Proceedings of the 29th Annual Symposium on User Interface Software
 1205 and Technology*. 29–39.
- 1206 [9] Xiang 'Anthony' Chen, Jeeeon Kim, Jennifer Mankoff, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2016. Reprise: A Design Tool
 1207 for Specifying, Generating, and Customizing 3D Printable Adaptations on Everyday Objects. In *Proceedings of the 29th Annual Symposium
 1208 on User Interface Software and Technology* (Tokyo, Japan) (*UIST '16*). Association for Computing Machinery, New York, NY, USA, 29–39.
 1209 <https://doi.org/10.1145/2984511.2984512>
- 1210 [10] Andrea S Ciullo, Janne M Veerbeek, Eveline Temperli, Andreas R Luft, Frederik J Tonis, Claudia JW Haarmann, Arash Ajoudani, Manuel G Catalano,
 1211 Jeremias PO Held, and Antonio Bicchi. 2020. A novel soft robotic supernumerary hand for severely affected stroke patients. *IEEE Transactions on
 1212 Neural Systems and Rehabilitation Engineering* 28, 5 (2020), 1168–1177.
- 1213 [11] Stelian Coros, Bernhard Thomaszewski, Gioacchino Noris, Shinjiro Sueda, Moira Forberg, Robert W Sumner, Wojciech Matusik, and Bernd Bickel.
 1214 2013. Computational design of mechanical characters. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.
- 1215 [12] Scott Davidoff, Nicolas Villar, Alex S Taylor, and Shahram Izadi. 2011. Mechanical hijacking: how robots can accelerate UbiComp deployments. In
 1216 *Proceedings of the 13th international conference on Ubiquitous computing*. 267–270.
- 1217 [13] Mark Edmonds, Feng Gao, Xu Xie, Hangxin Liu, Siyuan Qi, Yixin Zhu, Brandon Rothrock, and Song-Chun Zhu. 2017. Feeling the force: Integrating
 1218 force and pose for fluent discovery through imitation learning to open medicine bottles. In *2017 IEEE/RSJ International Conference on Intelligent
 1219 Robots and Systems (IROS)*. IEEE, 3530–3537.
- 1220 [14] Alberto Elfes. 1989. Using occupancy grids for mobile robot perception and navigation. *Computer* 22, 6 (1989), 46–57.
- 1221 [15] Erik D Engeberg, Nancy Pollard, Maximo Roa, and Zeyang Xia. 2018. Robotic grasping and manipulation competition: task pool. *Robotic Grasping
 1222 and Manipulation: First Robotic Grasping and Manipulation Challenge, RGMC 2016, Held in Conjunction with IROS 2016, Daejeon, South Korea,
 1223 October 10–12, 2016, Revised Papers* 816 (2018), 1.
- 1224 [16] T. Feix, J. Romero, H. B. Schmidtmayer, A. M. Dollar, and D. Kragic. 2016. The GRASP Taxonomy of Human Grasp Types. *IEEE Transactions on
 1225 Human-Machine Systems* 46, 1 (2016), 66–77. <https://doi.org/10.1109/THMS.2015.2470657>
- 1226 [17] David Fischinger, Peter Einramhof, Konstantinos Papoutsakis, Walter Wohlkinger, Peter Mayer, Paul Panek, Stefan Hofmann, Tobias Koertner, Astrid
 1227 Weiss, Antonis Argyros, et al. 2016. Hobbit, a care robot supporting independent living at home: First prototype and lessons learned. *Robotics and
 1228 Autonomous Systems* 75 (2016), 60–78.
- 1229 [18] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets robotics: The kitti dataset. *The International Journal of
 1230 Robotics Research* 32, 11 (2013), 1231–1237.
- 1231 [19] Yoav Golan, Amir Shapiro, and Elon Rimon. 2020. Jamming-free immobilizing grasps using dual-friction robotic fingertips. *IEEE Robotics and
 1232 Automation Letters* 5, 2 (2020), 2889–2896.
- 1233 [20] David Gyimothy and Andras Toth. 2011. Experimental evaluation of a novel automatic service robot tool changer. In *2011 IEEE/ASME International
 1234 Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 1046–1051.
- 1235 [21] Sehoon Ha, Stelian Coros, Alexander Alspach, James M Bern, Joohyung Kim, and Katsu Yamane. 2018. Computational design of robotic devices
 1236 from high-level motion specifications. *IEEE Transactions on Robotics* 34, 5 (2018), 1240–1251.
- 1237 [22] Alaa Hassan and Mohammad Abomoharam. 2014. Design of a single DOF gripper based on four-bar and slider-crank mechanism for educational
 1238 purposes. *Procedia CIRP* 21 (2014), 379–384.
- 1239 [23] Zhengtao Hu, Weiwei Wan, and Kensuke Harada. 2019. Designing a mechanical tool for robots with two-finger parallel grippers. *IEEE Robotics and
 1240 Automation Letters* 4, 3 (2019), 2981–2988.
- 1241 [24] Shohei Katakura, Yuto Kuroki, and Keita Watanabe. 2019. A 3D printer head as a robotic manipulator. In *Proceedings of the 32nd Annual ACM
 1242 Symposium on User Interface Software and Technology*. 535–548.
- 1243 [25] Shohei Katakura and Keita Watanabe. 2018. Printmotion: Actuating printed objects using actuators equipped in a 3D printer. In *The 31st Annual ACM
 1244 Symposium on User Interface Software and Technology Adjunct Proceedings*. 137–139.
- 1245 [26] Charles C Kemp, Aaron Edsinger, and Eduardo Torres-Jara. 2007. Challenges for robot manipulation in human environments [grand challenges of
 1246 robotics]. *IEEE Robotics & Automation Magazine* 14, 1 (2007), 20–29.
- 1247 [27] Frank Klassner and Scott D Anderson. 2003. Lego MindStorms: Not just for K-12 anymore. *IEEE robotics & automation magazine* 10, 2 (2003),
 1248 12–18.
- 1249 [28] Hikmet Kocabas. 2009. Gripper design with spherical parallelogram mechanism. *Journal of Mechanical Design* 131, 7 (2009).
- 1250 [29] Ioannis Kostavelis, Dimitrios Giakoumis, Georgia Peleka, Andreas Kargakos, Evangelos Skartados, Manolis Vasileiadis, and Dimitrios Tzovaras. 2018.
 1251 RAMCIP robot: a personal robotic assistant; demonstration of a complete framework. In *Proceedings of the European Conference on Computer Vision
 1252 (ECCV) Workshops*. 0–0.

- [30] Eric Krotkov, Douglas Hackett, Larry Jackel, Michael Perschbacher, James Pippine, Jesse Strauss, Gill Pratt, and Christopher Orlowski. 2017. The darpa robotics challenge finals: Results and perspectives. *Journal of Field Robotics* 34, 2 (2017), 229–240.
- [31] Kiju Lee, Yanzhou Wang, and Chuanqi Zheng. 2020. Twister hand: Underactuated robotic gripper inspired by origami twisted tower. *IEEE Transactions on Robotics* 36, 2 (2020), 488–500.
- [32] Jiahao Li, Meilin Cui, Jeeeon Kim, and Xiang'Anthony' Chen. 2020. Romeo: A Design Tool for Embedding Transformable Parts in 3D Models to Robotically Augment Default Functionalities. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 897–911.
- [33] Jiahao Li, Jeeeon Kim, and Xiang'Anthony' Chen. 2019. Robiot: A Design Tool for Actuating Everyday Objects with Automatically Generated 3D Printable Mechanisms. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 673–685.
- [34] Z-C Lia and C-H Meng. 1988. The dexterous workspace of simple manipulators. *IEEE journal on Robotics and Automation* 4, 1 (1988), 99–103.
- [35] Chih-Hsing Liu, Ta-Lun Chen, Chen-Hua Chiu, Mao-Cheng Hsu, Yang Chen, Tzu-Yang Pai, Wei-Geng Peng, and Yen-Pin Chiang. 2018. Optimal design of a soft robotic gripper for grasping unknown objects. *Soft robotics* 5, 4 (2018), 452–465.
- [36] Hangxin Liu, Chi Zhang, Yixin Zhu, Chenfanfu Jiang, and Song-Chun Zhu. 2019. Mirroring without overimitation: Learning functionally equivalent manipulation actions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 8025–8033.
- [37] Lucas Manuelli, Wei Gao, Peter Florence, and Russ Tedrake. 2019. kpam: Keypoint affordances for category-level robotic manipulation. *arXiv preprint arXiv:1903.06684* (2019).
- [38] R. Mutlu, G. Alici, M. In het Panhuis, and G. Spinks. 2015. Effect of flexure hinge type on a 3D printed fully compliant prosthetic finger. In *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. 790–795. <https://doi.org/10.1109/AIM.2015.7222634>
- [39] 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.
- [40] Ken Nakagaki, Yingda Liu, Chloe Nelson-Arzuaga, and Hiroshi Ishii. 2020. TRANS-DOCK: Expanding the Interactivity of Pin-based Shape Displays by Docking Mechanical Transducers. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction*. 131–142.
- [41] Kento Nakayama, Weiwei Wan, and Kensuke Harada. 2019. Designing grasping tools for robotic assembly based on shape analysis of parts. In *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–7.
- [42] Francesca Negrello, Werner Friedl, Giorgio Grioli, Manolo Garabini, Oliver Brock, Antonio Bicchi, Máximo A Roa, and Manuel G Catalano. 2020. Benchmarking hand and grasp resilience to dynamic loads. *IEEE Robotics and Automation Letters* 5, 2 (2020), 1780–1787.
- [43] H. Nishimura, A. Kakogawa, and S. Ma. 2012. Development of an underactuated robot gripper capable of retracting motion. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2161–2166. <https://doi.org/10.1109/ROBIO.2012.6491289>
- [44] Allison M Okamura, Niels Smaby, and Mark R Cutkosky. 2000. An overview of dexterous manipulation. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, Vol. 1. IEEE, 255–262.
- [45] Adam Pettinger, Conner Dimoush, and Mitch Pryor. 2019. Passive tool changer development for an elastic and compliant manipulator. In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 1200–1205.
- [46] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. Retrofab: A design tool for retrofitting physical interfaces using actuators, sensors and 3d printing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 409–419.
- [47] Iason Sarantopoulos, Yannis Koveos, and Zoe Doulgeri. 2018. Grasping flat objects by exploiting non-convexity of the object and support surface. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5606–5611.
- [48] Ashutosh Saxena, Justin Diemeyer, and Andrew Y Ng. 2008. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research* 27, 2 (2008), 157–173.
- [49] Philipp Schoessler, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. 2015. Kinetic blocks: Actuated constructive assembly for interaction and display. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 341–349.
- [50] Takeshi Takaki and Toru Omata. 2006. 100g-100N finger joint with load-sensitive continuously variable transmission. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 976–981.
- [51] Jun Ueda, Masahiro Kondo, and Tsukasa Ogasawara. 2010. The multifingered NAIST hand system for robot in-hand manipulation. *Mechanism and Machine Theory* 45, 2 (2010), 224–238.
- [52] Chen Wang, Daniel Freer, Jindong Liu, and Guang-Zhong Yang. 2019. Vision-based Automatic Control of a 5-Fingered Assistive Robotic Manipulator for Activities of Daily Living. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 627–633.
- [53] Zeli Wang, Heng Li, and Xintao Yang. 2020. Vision-based robotic system for on-site construction and demolition waste sorting and recycling. *Journal of Building Engineering* 32 (2020), 101769.
- [54] Jingren Xu, Weiwei Wan, Keisuke Koyama, Yukiyasu Domae, and Kensuke Harada. 2021. Selecting and designing grippers for an assembly task in a structured approach. *Advanced Robotics* (2021), 1–17.
- [55] Simon Zimmermann, Ghazal Hakimifard, Miguel Zamora, Roi Poranne, and Stelian Coros. 2020. A multi-level optimization framework for simultaneous grasping and motion planning. *IEEE Robotics and Automation Letters* 5, 2 (2020), 2966–2972.