HardwareX 1 (2017) 13–21

Contents lists available at ScienceDirect

HardwareX

journal homepage: www.elsevier.com/locate/ohx

Hardware Article

An open source automatic feeder for animal experiments

Jinook Oh [*,1], Riccardo Hofer [1], W. Tecumseh Fitch

Cognitive Biology Department, University of Vienna, Althanstrasse 14, 1090 Vienna, Austria

ARTICLE INFO

Article history:
Received 26 June 2016
Received in revised form 6 September 2016

Keywords:
Automatic feeder
Open source hardware
Animal experiments
Operant conditioning

ABSTRACT

Automatic feeders are widely used in animal experiments to dispense an accurate amount of food reward for each trial. Several commercial automatic feeders for animal experiments are available which are specific to certain species and food types. However, it would be beneficial for researchers if they could easily build their own experimental feeders customized for their study species, food types, and other experimental considerations. In this paper, we describe an open source experimental feeder using an Arduino microcontroller. The design of the feeder is focused on simplicity to provide a straight-forward building process and allow custom modifications for various requirements. The cost for building this feeder is less than €200 and we have successfully tested our design with three different food types for pigeons, cats, and marmoset monkeys.

1. Introduction

In cognitive biology, psychology, neuroscience and other disciplines studying animal behavior, many experiments have been conducted using automatic machine feeders [1–4] to coax animals to participate in trials and provide food rewards to facilitate learning processes. Commercial automatic feeders for animal experiments from several companies are widely used. However, each commercial feeder type is specifically designed for a particular animal species and/or a food type. Furthermore, these devices are independently designed without knowledge of any specific animal experiment or experimental paradigm. Fully customized systems such as [5–7] have also been built and used in animal experiments, however, most existing custom solutions are entirely or partially closed-source and again specifically designed for a specific species and/or experiment type. Also, descriptions of hardware components even in open source systems are not systematically described in detail for other researchers to build one. Many animal experiments have specific requirements depending on subject species, specifications of experimental apparatus and other experimental details. Therefore, adjustments to the experimental apparatus or purchases of new devices are often required. Making a device for animal experiments as an open source project would be beneficial to many researchers, especially considering the need for frequent adjustments as well as a desire for reduced cost. To our knowledge, the benefits of open source custom devices have not yet been fully exploited in animal studies, perhaps because building a machine was considered to be a difficult and time-consuming task for biologists, psychologists and other researchers focused on the study of animal behavior. Recently, building a custom machine has become a much more feasible task, thanks to inexpensive, open source microcontrollers, laser cutters, 3D printers and other recently developed "maker" technologies.

* Corresponding author.
E-mail addresses: jinook.oh@univie.ac.at (J. Oh), riccardo.hofer@univie.ac.at (R. Hofer), tecumseh.fitch@univie.ac.at (W.T. Fitch).
[1] These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.ohx.2016.09.001
2468-0672/

This paper describes an open source automatic feeder for animal experiments. This feeder can be used for most sizes of dry food pellets with the potential modification of a single component; also, it can be built at a low cost by researchers without detailed knowledge of engineering. We built and tested the feeder for three different food types, seeds for pigeons (*Columba livia*), pellets for cats (*Felis catus*) and pellets for common marmoset monkeys (*Callithrix jacchus*), showing that the design is applicable to a range of animal species.

2. Results

All the necessary files to build the feeder including schematic drawing, diagrams, arduino code and testing video files are stored in Open Science Framework (`osf.io/j57dp`) and Github (`github.com/jinook0707/openfeeder`). The feeder, shown in Fig. 1, was tested with three different types of foods shown in Fig. 2. We tested the feeder with each food type for 1000 dispenses and the result is depicted in Fig. 3.

In testing seeds for pigeons, eight errors (no or negligible amount of food was dispensed) occurred because small seeds were temporarily stuck around the dispensing hole and dropped onto the piezo-electric sensor at an inappropriate time. One to two pellets for cats were dispensed at a time. No specific error in cat pellets occurred during this test. In the error cases of pellets for marmoset monkeys, the feeder dispensed two pellets at a time because the piezo-electric sensor failed to sense an impact of a pellet due to its small size and the location of an impact. This error occurred 12 times in total.

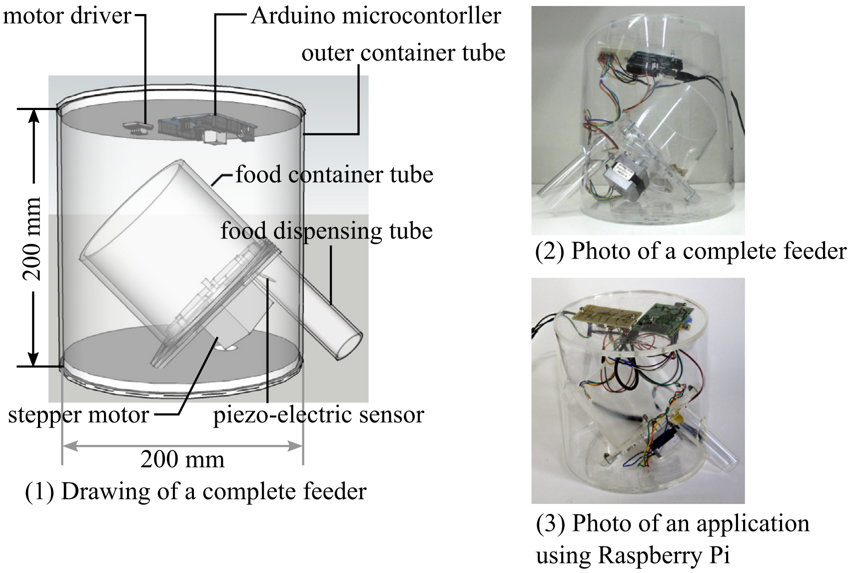

Fig. 1. Complete feeders.

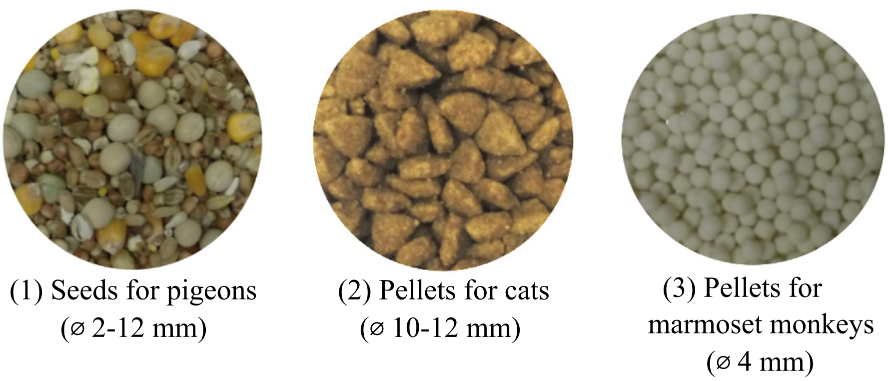

Fig. 2. Three tested food types for three species.

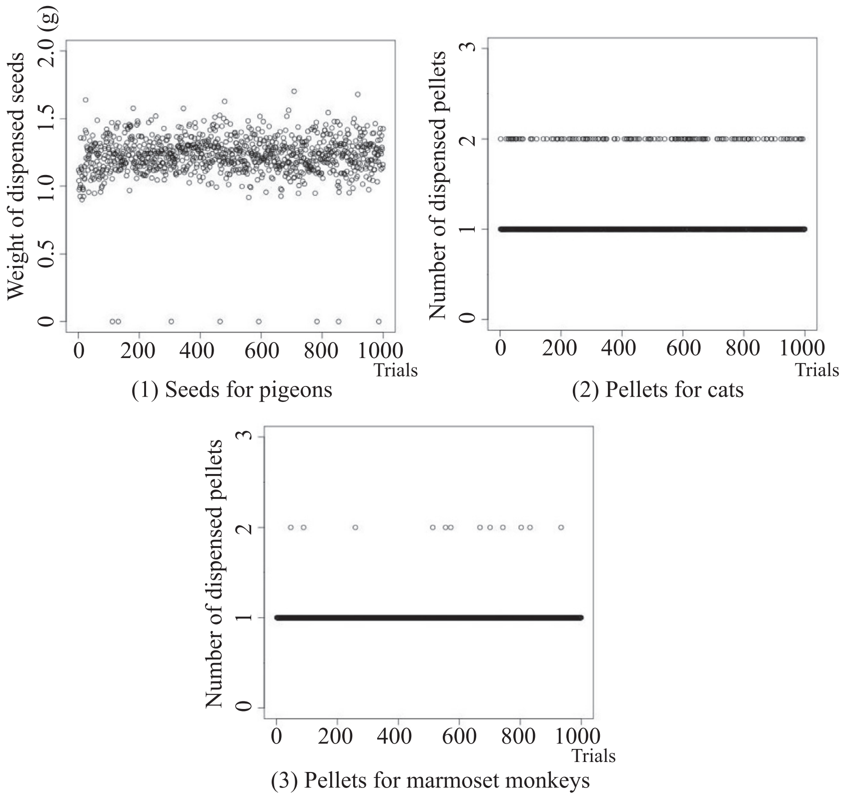

Fig. 3. Test result of the feeder.

3. Materials and methods

3.1. Materials

The following materials were used to build the automatic feeder.

- *c1* plexiglass tube with dimensions, 200 mm (OD; Outer Diameter)/194 mm (ID; Inner Diameter)/200 mm (H; Height)
- *c2* plexiglass tube with dimensions, 120 mm (OD)/110 mm (ID)/150 mm (H)
- *c3* plexiglass tube with dimensions, 30 mm (OD)/26 mm (ID)/100 mm (H)
- *c4* Acrifix cement
- *c5* plexiglass XT parts from a laser cutting company (Fig. 4)
- *c6* Mercury SM-42BYG011-25 stepper motor
- *c7* Pololu mounting disk with dimensions, 19 mm (OD)/5 mm (H; Height) (This disk has M2.5 holes. We tapped them to use M3 bolts.)
- *c8* Arduino Uno
- *c9* permanent breadboard
- *c10* piezo electric sensor (2 cm)
- *c11* power adapter
- *c12* USB cable
- *c13* Pololu A4988 stepper motor driver

3.2. Procedures to build the feeder

3.2.1. Drawing for plexiglass laser cutting

Lines for cutting plexiglass parts (*c4*) were drawn (Fig. 4) using a vector graphic software and sent to a laser cutting company to order the required plexiglass parts. This laser cutting procedure can be done with a shared laser cutter if it is available. Notches of (d) should be adjusted to the size and amount of foods to deliver for one dispense event. For seeds and cat food pieces, we needed a thicker disk (d) to deliver enough food pieces. We made three of the disks (d) and glued them together using Acrifix (*c4*) instead of using one 9 mm thick plexiglass plate. As a plexiglass plate becomes thicker, an error

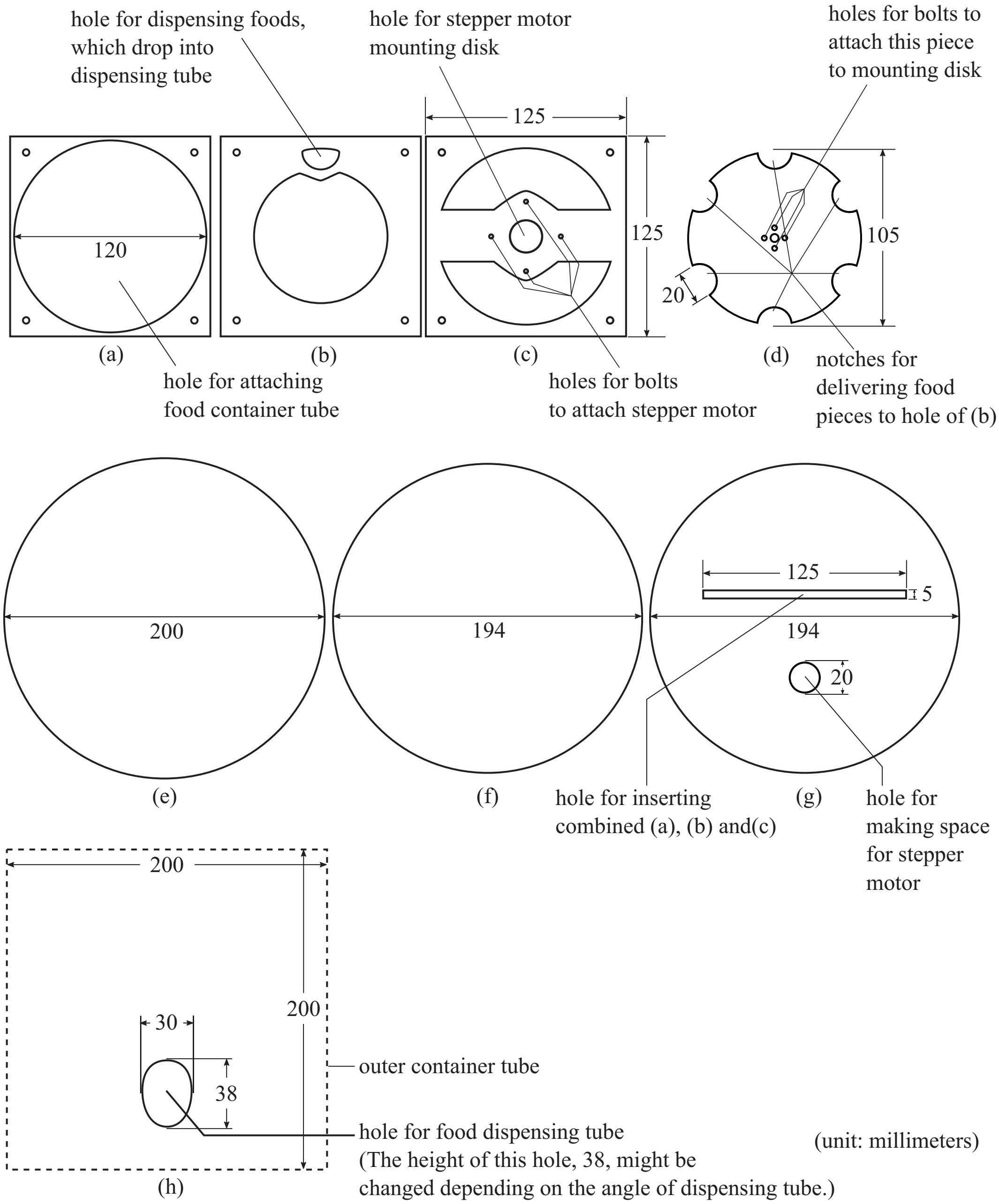

Fig. 4. Drawing for laser cutting; (f) and (g) are 6 mm thick and other pieces are 3 mm thick.

in laser cutting usually becomes larger because the power of the laser decreases as it penetrates a thick plexiglass plate. We tried to reduce this type of error especially in (d), considering its crucial role of delivering food pieces toward the dispensing tube. Two of (e) in Fig. 4 were ordered to form a top plate with (f) and a bottom plate with (g). Only the oval shaped hole in (h) is for laser cutting; the square with dotted lines represents the outer container tube.

3.2.2. Wire electronics

Electronics were prepared by wiring as shown in Fig. 5.

3.2.3. Assembly of plexiglass components and electronic components

The food container tube (*c2*) was cut to a desired length (10 cm in our tests) and glued to (a) in Fig. 4. The parts (a), (b), and (c) were combined together. The mounting disk was attached to the shaft of the stepper motor; then the motor was

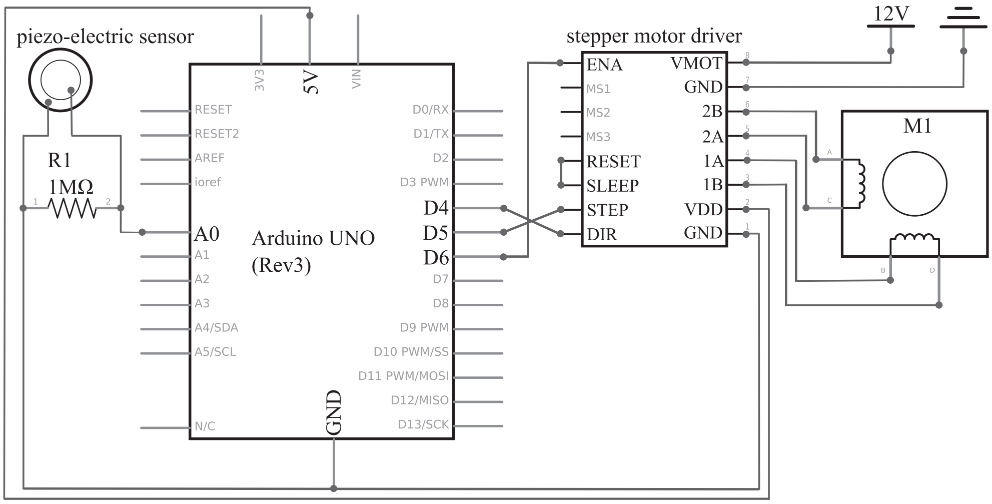

Fig. 5. Schematic drawing of electronic parts.

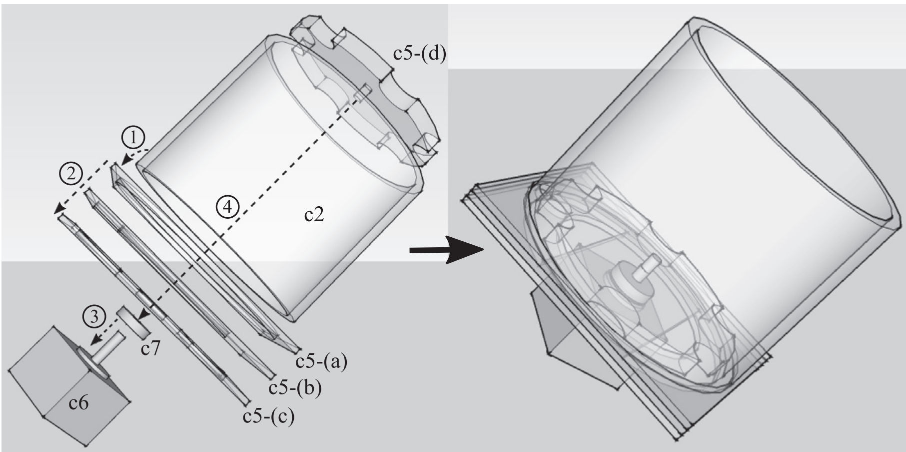

Fig. 6. Assembly of the core part; *c2*, *c5*, *c6* and *c7* are described in the Section 3.1; circled numbers represent order of assembly.

attached to (c). The food delivering disk (d) was mounted on the mounting disk. The assembled components to this point make up the core part of this feeder (Fig. 6). Bolts, nuts and washers were used to assemble the core part of the feeder, unless mentioned otherwise. Components close to the stepper motor were assembled with plastic bolts, nuts and washers to reduce noise and vibration caused by motor rotations.

To measure the effect of the plastic connection components (plastic bolts, nuts and washers), especially the four bolts to connect (d) and (*c7*), we recorded noise from motor operations with the assembled core part. The assembled core part was positioned on a frame made of metal beams as shown in Fig. 7(1). A Zoom H4N recorder was placed approximately 15 cm away from the core part and recorded sounds while the motor was rotating. Its sound pressure levels on frequencies are drawn in Fig. 7(2–4) using Praat [8]. Also, maximum sound pressure levels in each case were measured using Voltcraft SL-451 SPL meter. The maximum sound pressure levels in normal room noise, motor operation with plastic components and motor operation with metal components were 52 dB(C), 65.4 dB(C) and 76.9 dB(C) respectively.

The parts (e) and (f) were glued together to form a top plate and (e) and (g) were glued together to form a bottom plate. The Arduino and the motor driver were fixed to the top plate after drilling several holes. The core part was laid on top of the bottom plate and the outer container tube (c1) was positioned to surround the core part. The dispensing tube (c3) was cut to a desired length (10 cm in our tests) and a slit at one end of the tube was cut with a hand saw. The wire attached to the piezo-electric sensor was positioned in the slit and the sensor was firmly fixed with glue as shown in Fig. 8. The drawing (1) of Fig. 8 shows that the sensor is not completely aligned with the line of the dispensing tube; this made the sensor more sensitive to an impact of a small and light food piece. During our tests, this alignment was necessary only for the pellet for the marmoset monkeys due to its light weight. The dispensing tube was positioned through the hole in (h) of Fig. 4 and fixed

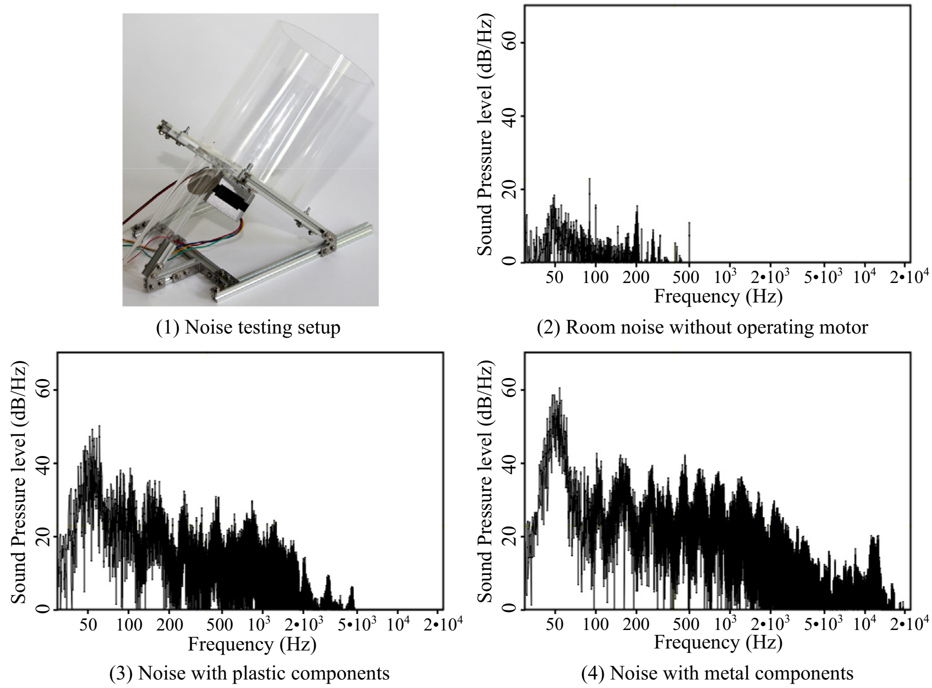

Fig. 7. Intensity of noise.

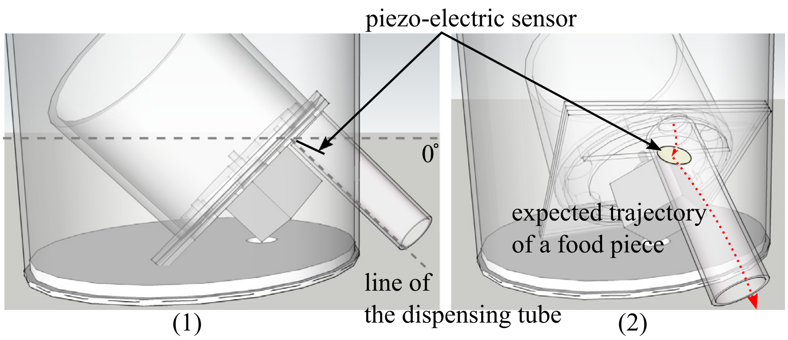

Fig. 8. Placing piezo-electric sensor and dispensing tube.

with Acrifix. The dispensing tube should not touch the motor to avoid sensing vibrations of the motor because the Arduino microcontroller could interpret these vibrations as a dropped food piece (Fig. 9).

3.2.4. Arduino code

A user can download the arduino code from Open Science Framework (`osf.io/j57dp`) or Github(`github.com/jinook0707/openfeeder`). Arduino software and AccelStepper library is required to upload this code to Arduino microcontroller. Arduino software can be downloaded from https://www.arudino.cc and installed in a straight- forward manner. AccelStepper library can be downloaded from http://www.airspayce.com/mikem/arduino/AccelStepper/index.html. Installing arduino library is also easy to follow as shown in their guide page https://www.arduino.cc/en/Guide/Libraries. After opening the code with Arduino software, board type and a serial port should be specified under 'Tools' menu. Then, the code can be uploaded by pressing 'Upload' button. More guides about Arduino software can be found in https://www.arduino.cc/en/Guide/HomePage. The algorithm of 'loop' function in our code is depicted in Fig. 10.

'Distance to rotate' is a constant in the code. A user does not have to change this constant each time they slightly change the design as long as it is large enough to reach the next food dispensing notch. The time to stop the stepper motor is determined by food piece's impact on the piezo-electric sensor, not by specified distance or time. 'Rotating motor back and forth with increased speed' in Fig. 10 was implemented for the case when a food piece was stuck on the rotating disk. This operation successfully removed stuck food pieces in our testing sessions.

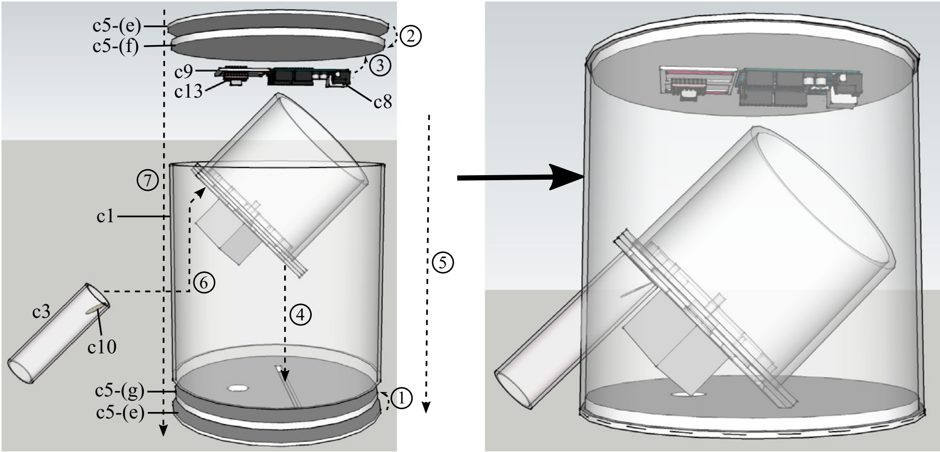

Fig. 9. Assembly of all the parts; *c1*, *c3*, *c5*, *c8*, *c9*, *c19* and *c13* are described in Section 3.1; circled numbers represent order of assembly.

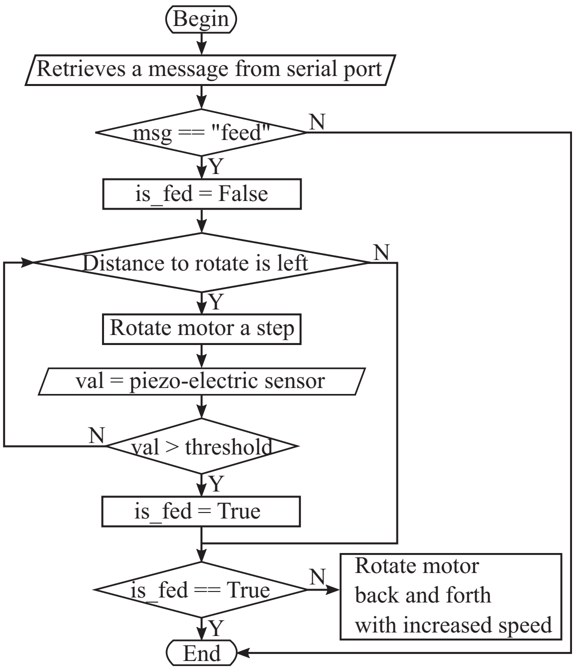

Fig. 10. Flowchart of 'loop' function of Arduino code. The 'loop' function is constantly called while Arduino microcontroller is powered on, therefore, it is the main part of our code.

3.3. Required time frame for overall building procedure

Five to six hours were required to build one feeder. This time of assembly does not include time for purchase, delivery or laser cutting processes. When a technically well informed researcher tries to build this feeder for the first time, this assembly time might increase from several hours to several days depending mainly on the researcher's experience in electronics. However, this required time will quickly decrease within several building experiences.

4. Discussion

Regarding novelty, cost and availability, we do not claim this feeder is the first, cheapest or easiest one to obtain. In animal behavior research fields, there have been exemplar open source systems [9,10] to train animal species. These systems include custom feeder devices, however, they do not provide clear enough descriptions to build hardware components for researchers in non technological fields. Unlike the previous open source system reports, we believe that this report provides enough information for a researcher in non technological field to build an adjustable automatic feeder without external technical support. Also, we are aware of and have used several commercial automatic feeders in our department. In certain cases, we had to physically and/or procedurally adjust our training systems to use the commercial feeder. In other cases, we had to hack the commercial product to control it with our software via a microcontroller. A series of these modifications led us to develop a feeder which was an adjustable base unit for different setups, open source and relatively easy to build. We believe that open source tools are not yet fully exploited in animal behavior research fields and more official reports including detailed descriptions of building processes are required to make open source tools feasible for researchers in the fields.

The feeder shown in Fig. 1 dispensed approximately 180 times over 12 h per day as it fed two cats for over four months. Another feeder was used within a pigeon training device for several weeks as Fig. 11(1). A third one was adjusted to a monkey training device (Fig. 11) and has been used for two months. These three test cases showed that the developed feeder can be employed for long term usage and also be easily adjusted to be a part of other devices thanks to its simple design.

There was some concern about food dust buildup resulting from rubbing between feeder parts and food pieces. The seeds for pigeons were hard and did not produce much of dust from rubbing. The pellets for marmoset monkeys were BioServ's 'Dustless Precision Pellets, F0059', which did not produce dust either as its name suggested. The pellets for cats indeed produced dust from rubbing. The left picture of Fig. 12 shows the core part of our prototype feeder after 20,000 dispenses over a four month period, which directly rubs on food pieces. The right picture shows the core part of the currently described feeder after 3500 dispenses over three weeks. The only difference between these two feeders is the type of food container tube and

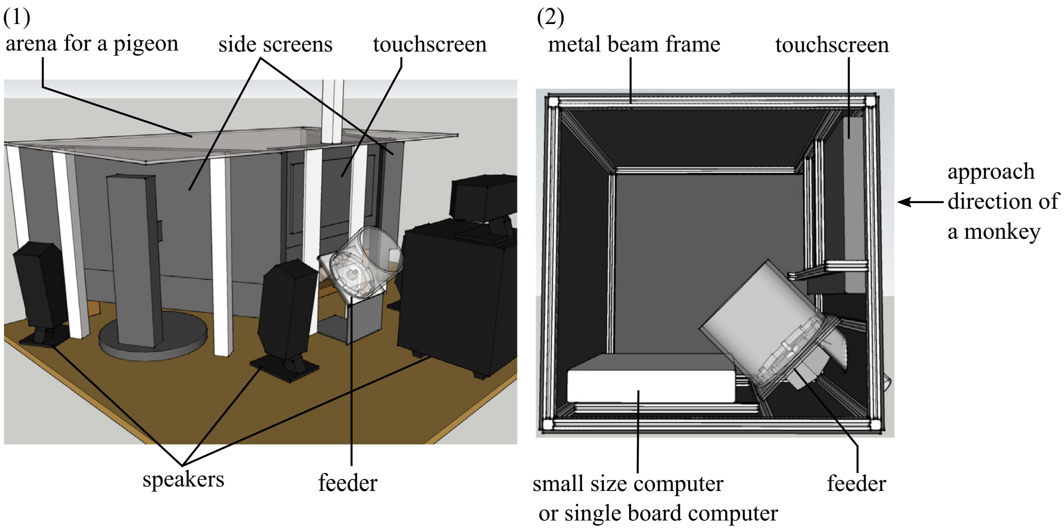

Fig. 11. Adjusted feeders for other devices.

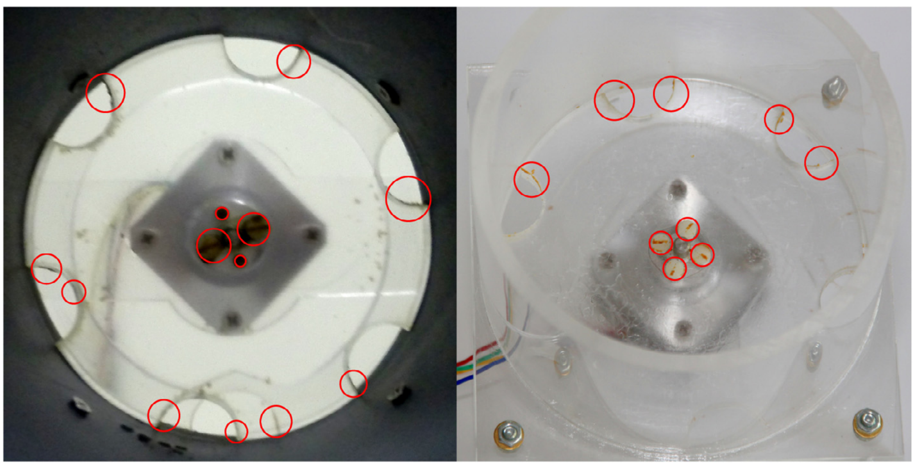

Fig. 12. Photos of food dust residue after 20,000 and 3500 dispenses respectively; circles indicate locations of built up dust.

how we fixed it. One with bolts and the other one with Acrifix. Both pictures were taken without any pre-cleaning. Some food dust built up on the disk, bolts and other parts of the feeder, however, it was not significant enough to reduce its performance. Even if it eventually builds up to reduce its performance after several months, it will not cause a serious problem because a user who built this feeder from scratch can also quickly disassemble all the parts to clean it up. This is one of the main benefits of using open source hardware, i.e. one can assemble, disassemble and modify without much difficulty.

We will further develop the current feeder to build several different versions. One is a 3D printed version replacing most plexiglass components to a singular 3D printed frame to simplify the assembly process. Others are extended versions implementing sensors and actuators via the microcontorller to extend its functionality.

Authors' contributions

J.O. and R.H. conceived the feeder. J.O. wired electronics, wrote Arduino code and this manuscript. R.H. designed and produced the feeder except electronic parts. T.F. gave advice for the overall procedure and revised this manuscript.

References

[1] U. Aust, L. Huber, The role of item-and category-specific information in the discrimination of people versus nonpeople images by pigeons, Animal Learn. Behav. 29 (2001) 107–119.
[2] S. Liu, R.P. Heitz, C.W. Bradberry, A touch screen based stop signal response task in rhesus monkeys for studying impulsivity associated with chronic cocaine self-administration, J. Neurosci. Methods 177 (2009) 67–72.
[3] N. Stobbe, G. Westphal-Fitch, U. Aust, W.T. Fitch, Visual artificial grammar learning: comparative research on humans, kea (*Nestor notabilis*) and pigeons (*Columba livia*), Philos. Trans. Roy. Soc. Lond. B Biol. Sci. 367 (2012) 1995–2006.
[4] J.C. Talpos, N. Aerts, L. Fellini, T. Steckler, A touch-screen based paired-associates learning (PAL) task for the rat may provide a translatable pharmacological model of human cognitive impairment, Pharmacol. Biochem. Behav. 122 (2014) 97–106.
[5] J. Fagot, E. Bonté, Automated testing of cognitive performance in monkeys: use of a battery of computerized test systems by a troop of semi-free-ranging baboons (Papio papio), Behav. Res. Methods 42 (2010) 507–516.
[6] M.M. Steurer, U. Aust, L. Huber, The Vienna comparative cognition technology (VCCT): an innovative operant conditioning system for various species and experimental procedures, Behav. Res. Methods 44 (2012) 909–918.
[7] L. Huber, N. Heise, C. Zeman, C. Palmers, The ALDB box: automatic testing of cognitive performance in groups of aviary-housed pigeons, Behav. Res. Methods 47 (2015) 162–171.
[8] P. Boersma, D. Weenink, Praat, a system for doing phonetics by computer, Glot Int. 5 (2001) 341–345.
[9] O. Pineño, Arduipod box: a low-cost and open-source skinner box using an ipod touch and an arduino microcontroller, Behav. Res. Methods 46 (2014) 196–205.
[10] J. Oh, W.T. Fitch, Catos (Computer Aided Training/Observing System): automating animal observation and training, Behav. Res. Methods 1 (2016).