Journal of open hardware

A Precise Dispenser Design for Canine Cognition Research

HARDWARE METAPAPER

]u[ubiquity press

WALKER ARCE

JEFFREY R. STEVENS

*Author affiliations can be found in the back matter of this article

ABSTRACT

Some forms of canine cognition research require a dispenser that can accurately dispense precise quantities of treats. When using off-the-shelf or retrofitted dispensers, there is no guarantee that a precise number of treats will be dispensed. Often, they will over-dispense treats, which may not be acceptable for some tasks. Here we describe a 3D-printed precise treat dispenser with a 59-treat capacity driven by a stepper motor drive and controlled by an integrated Raspberry Pi. The dispenser can be built for less than 200 USD and is fully 3D printable. While off-the-shelf dispensers can result in an error rate of 20–30%, the precision dispenser produces a 4% error rate. This lower error rate and the integrated Raspberry Pi allows for new possibilities for using treat dispensers across a range of canine research questions.

METADATA OVERVIEW

Main design files: *https://github.com/unl-cchil/canine_precise_dispenser*

Target group: Scientists in canine cognition.

Skills required: desktop 3d printing – easy; through-hole soldering – easy; hand tool use – easy

Replication: No builds known to the authors so far.

See section "Build Details" for more detail.

CORRESPONDING AUTHORS:

Walker Arce

Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, US

warce@unomaha.edu

KEYWORDS:

Canine; computer-controlled dispenser; numerical cognition; 3D printed; Raspberry Pi; hardware

TO CITE THIS ARTICLE:

Arce, W., & Stevens, J. R. 2022. A Precise Dispenser Design for Canine Cognition Research. *Journal of Open Hardware*, 6(1): 2, pp. 1–13. DOI: *https://doi.org/10.5334/joh.41*

(1) OVERVIEW

INTRODUCTION

Canine research uses a variety of methods to investigate dog behavior and cognition (Miklósi 2015). Many of these methods involve dispensing food treats to dogs based on choices that they have made. Dispensing the treats often requires elaborate experimental designs, using a series of tubes and a human present to dispense the treats (e.g., Utrata et al. 2012). While these designs may work in some situations, they can also lead to bias from human presence, introduce delays in reward timing, or limit where the studies can take place. Previously, we developed a system to automate treat dispensing by retrofitting an off-the-shelf dispenser with an integrated Raspberry Pi (Arce & Stevens 2020). Due to the precision of the off-the-shelf dispenser, this system has an error rate of 20–30%, resulting in dispensing more than one treat at a time. This system works well for canine research projects that do not rely on precise control over the number of treats dispensed.

Some canine research, however, requires precise control over the number of treats dispensed. For instance, studies on numerical cognition or how well animals can understand quantities of objects (Davis & Perusse 1988; Brannon 2006) may require that an individual receive a precise number of treats based on their choice. If they are meant to receive three treats and they accidentally receive four, that can be problematic for the study. Therefore, we need devices that can automatically dispense precise numbers of treats. By developing a standard platform for precision treat dispensing, future studies can replicate dispensing methods and reduce variability in the experimental setup.

In looking at other dispenser designs, many have a central disc that has cells equally spaced along its perimeter that contain the media to be dispensed (Abdullahi et al. 2019, Robinson et al. 2020, Thomas et al. 2021, Wallis et al. 2017). This disc rotates around its center and each cell will go over an opening and fall into the receptacle where the dog will consume the treat. The dispenser presented by Abdullahi et al. has a resizable disc that can dispense treats from a wheelchair mount, but the small capacity limits its applicability to numerical discrimination studies. The dispenser presented by Robinson et al. was portable but had a limited capacity for treats. The dispenser presented by Thomas et al. seemed to have a large capacity, it is portable, but it has a launching mechanism, which was not needed in our design. The dispenser presented by Wallis et al. had an acceptable capacity but was not portable. Dispensers for other subjects exist, such as fruit fly dispensers, which integrate accurate mechanical platforms to move a dispensing peristaltic pump over fruit fly containers (Wayland & Landgraf 2018).

Our previous work retrofitted the *Treat & Train* (PetSafe 2020) dispenser to provide a solid software framework for creating, delivering, and managing experimental setups (Arce & Stevens 2020). The new dispenser design needs to address the shortfalls of the retrofitted *Treat & Train*. First, dispensing a single treat needs to have a high rate of success, which means that chaining together dispensing events must have a low rate of over- or under-dispensing treats. Second, the pellet exhaust needs to have a method by which the treat being dispensed can be measured and detected to maintain a precise count of treats dispensed. Third, the benefit of devices like the *Treat & Train* is the large reservoir of treats that can be accessed for the experiment, which leaves little room for error from running out of treats. The developed dispenser needs to have a high enough treat capacity to facilitate longer experiments. Considering these previous designs, we decided to proceed with the dispensing wheel from Abdullahi et al. and Wallis et al., the portability and local computing from Robinson et al., the accurate mechanical movement using a stepper motor from Wayland & Landgraf, and the capacity of Thomas et al.

Our aim here is to describe the development of the method of construction, operation, and reliability of a device that is a self-contained experimental system that can be used in canine research studies. To do this, we designed a 3D printable dispenser with a single treat dispensing architecture that integrates the standard features for a dispenser: treat dispensing detection, large treat capacity, and isolation of treats from the experiment subject. The device is low enough in cost to be accessible to canine cognition researchers. It is also simple enough to assemble that only standard relevant skills are necessary, such as operation of a 3D printer and a soldering iron. In addition, we integrate a Raspberry Pi 4, which allows researchers to display output to a monitor, receive input from a mouse, keyboard, and touchscreen, and then control the dispenser. Therefore, this design offers a self-contained system that can be used to run all aspects of a canine cognition research project using, for instance, the Python programming language to control all components.

OVERALL IMPLEMENTATION AND DESIGN

Hardware Development

The precision dispenser comprises two major components: the mechanical and electrical components. Other than the Raspberry Pi, motor, and acrylic cover, each piece was designed in Autodesk Fusion 360 (Autodesk 2021) and 3D printed with a Sovol SV01. The version documented in this paper is v84, with the f3z files available in the project repository, which can be imported directly into Fusion 360.

An exploded view shows how the pieces come together, using 27 M3 screws (*Figure 1*).

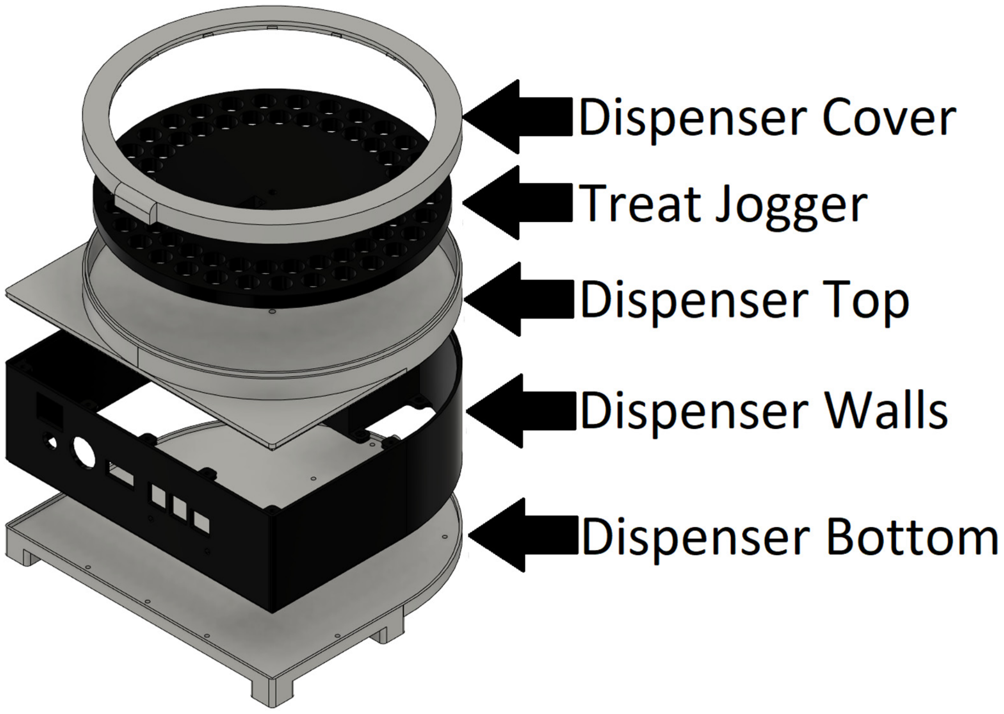


Figure 1 Exploded view of the dispenser isolating its mechanical components.

The cover goes over the top of the dispenser and seals the treats from the canine in the experiment. The cover consists of a 3D printed rim and an eight-inch acrylic disk set in a recess on the underside of the rim to fully seal off the treats from the external environment (*Figure 2*). The cover rim includes a tab to allow it to be easily removed from the dispenser.

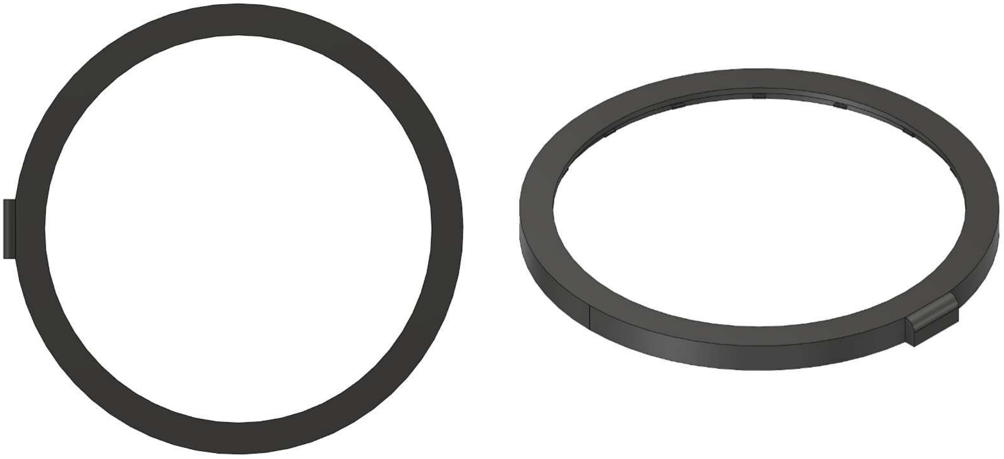


Figure 2 Dispenser cover with a center depression for an 8-inch acrylic disk.

The treat jogger consists of a 3D printed wheel with two circles of receptacles for treats, leading to a total treat capacity of 59 treats (*Figure 3*). The holes are sized for the Pet Botanics Training Reward treats, so the holes are 15 mm. The shaft of a standard 1.8°, Nema17 stepper motor is set in the center of the wheel, with a set screw used to solidly affix the wheel to the motor shaft.

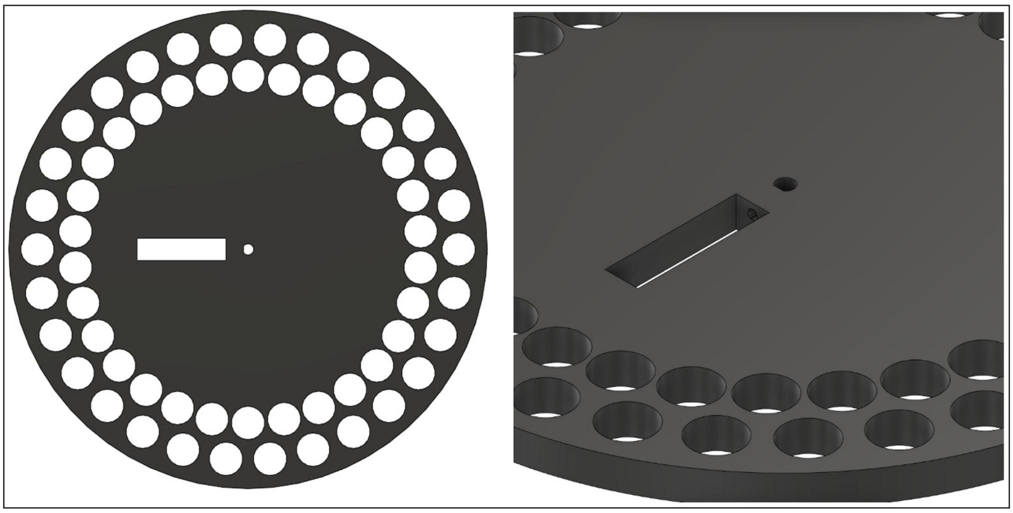


Figure 3 (Left) Overhead view of the treat jogger and (right) a detail view of the set screw.

The top plate of the dispenser has the stepper motor mounted to it and the top of the pellet exhaust exposed, which is elongated to allow either the top or bottom row treat to be dispensed (*Figure 4*). The top of this plate has a lip to press fit the dispenser cover and the bottom of this plate has a lip to press fit the walls as well as complimentary M3 screw holes to securely attach the two pieces.

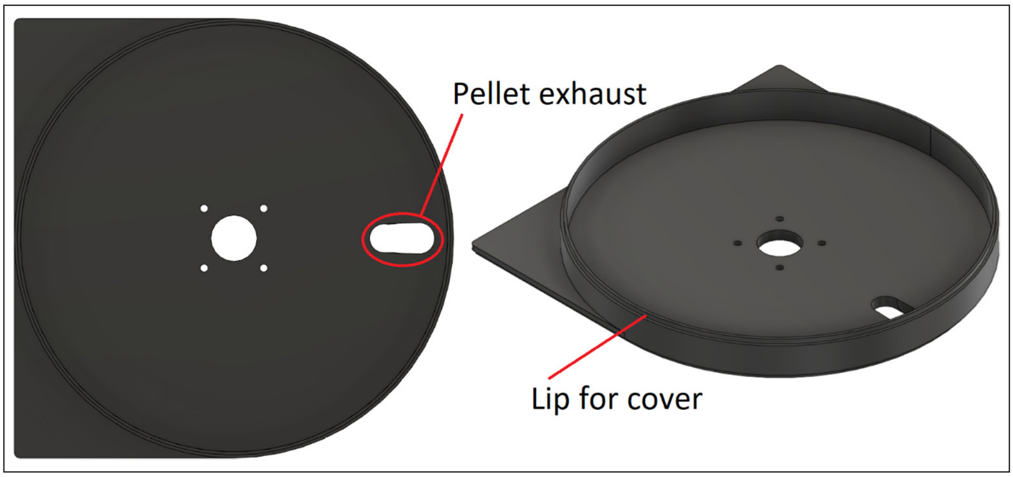


Figure 4 Dispenser top plate. (Left) Overhead view of the top plate of the dispenser and (right) a detailed view showing the two lips of the plate.

The walls of the dispenser have the electronics mounted to it and exposes the connectors for power and data (*Figure 5*). The pellet exhaust can be seen as well as the mounting point for the infrared (IR) break beam sensor, which has a complimentary detector on the opposite side of the pellet exhaust. Each of the bottom mounting points for the bottom plate have an M3 nut capture to allow the bottom plate to be screwed on without holding the nut in place.

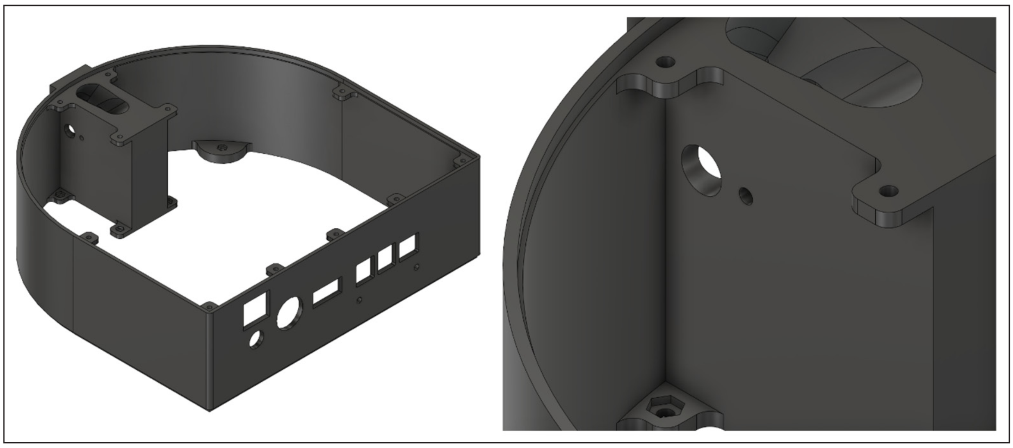


Figure 5 Dispenser walls. (Left) The walls of the dispenser which has the connectors and electronics mounted to it and (right) a detail shot of the mounting point for the IR break beam sensor.

The bottom plate of the dispenser closes the system up and isolates the electronics from the main experiment area (*Figure 6*). Eleven M3 screws secure the bottom plate to the walls of the dispenser. This is also the access point for service to the electronics.

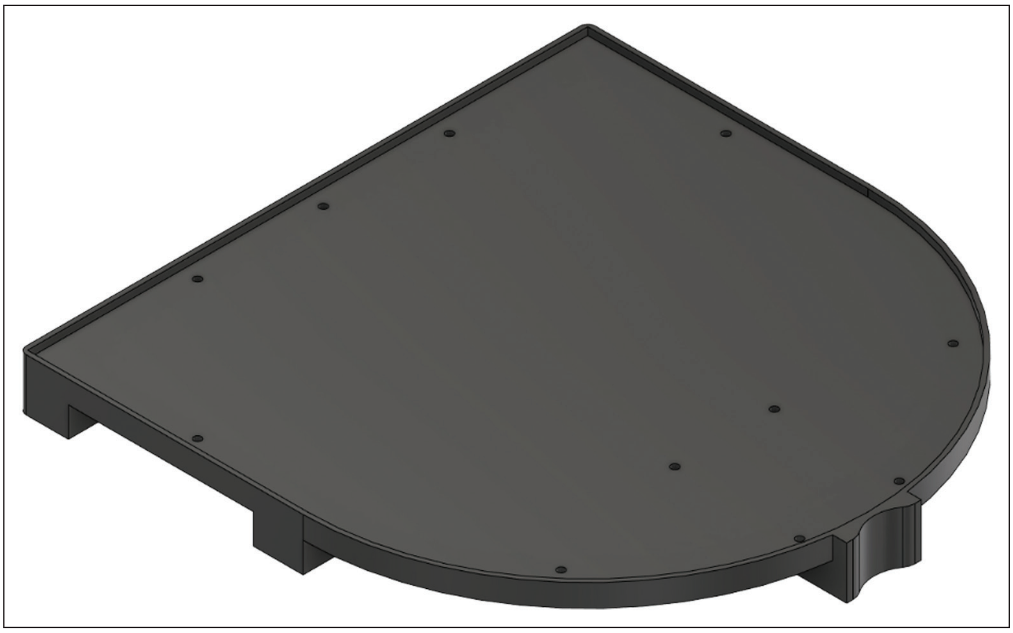


Figure 6 Bottom plate of the dispenser with feet to stand it higher off the main surface.

The connector panel of the dispenser exposes all connection points necessary to interact with the system (*Figure 7*). The two power supplies can be disconnected from the system using the double-pull, double-throw (DPDT) power switch, a display can be connected through the HDMI connector, and USB and Ethernet can be accessed directly on the Raspberry Pi.

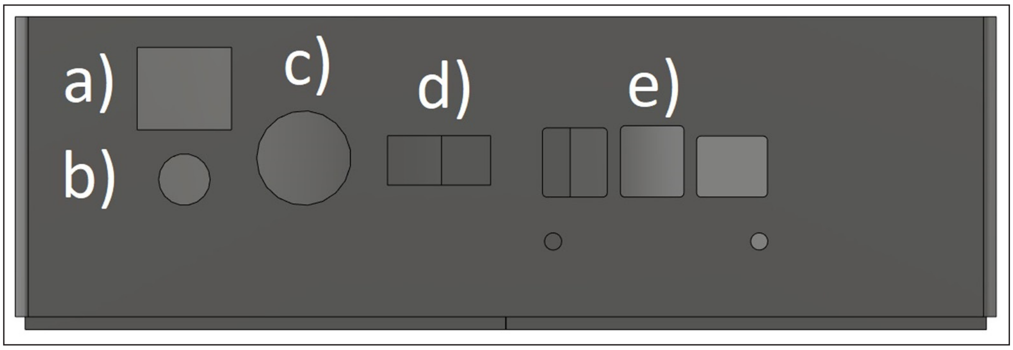
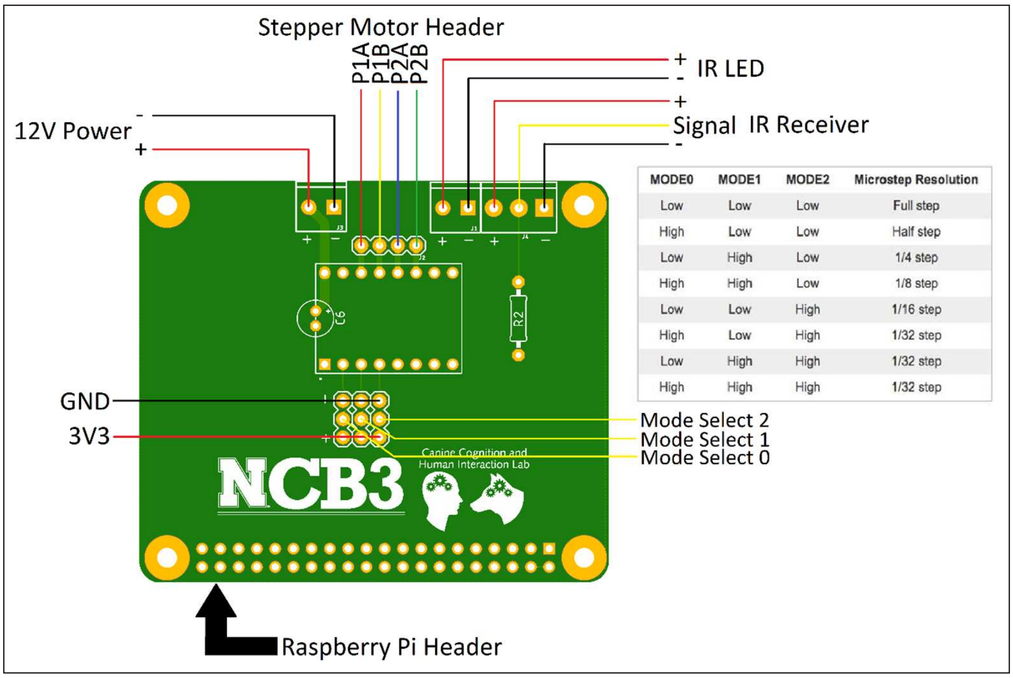


Figure 7 Connector panel detail view. (a) global power switch, (b) 12V motor power supply, (c) 5V Raspberry Pi power supply, (d) HDMI panel mount connector, (e) exposed Raspberry Pi 4 connectors.

Finally, the custom Raspberry Pi HAT (Hardware Attached on Top) (*Figure 8*) mounts the stepper motor driver and screw terminals for the IR break beam and stepper motor power source. The



MODE0	MODE1	MODE2	Microstep Resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	1/4 step
High	High	Low	1/8 step
Low	Low	High	1/16 step
High	Low	High	1/32 step
Low	High	High	1/32 step
High	High	High	1/32 step

Figure 8 Raspberry Pi HAT.

Pi HAT was designed using Autodesk Eagle (Autodesk, 2021). The Pi HAT can be assembled using just a soldering iron as all the parts are through-hole, making assembly simple. The stepper motor driver that was used in our setup was the A4988 stepper motor driver carrier. By soldering female headers onto the stepper driver footprint, as per the bill of materials, the stepper motor driver can be replaced easily (*Figure 9*). Any 3D printer compatible stepper motor driver carrier will work. The mode select pins control the stepping resolution of the motor; for our setup, we used 1/16 stepping resolution. The current limit will also need to be set for each driver, which can be accomplished by following Pololu Robotics and Electronics' tutorial (Pololu, 2015). The phototransistor connects to the right-side terminals, which goes low when a treat breaks the IR beam.

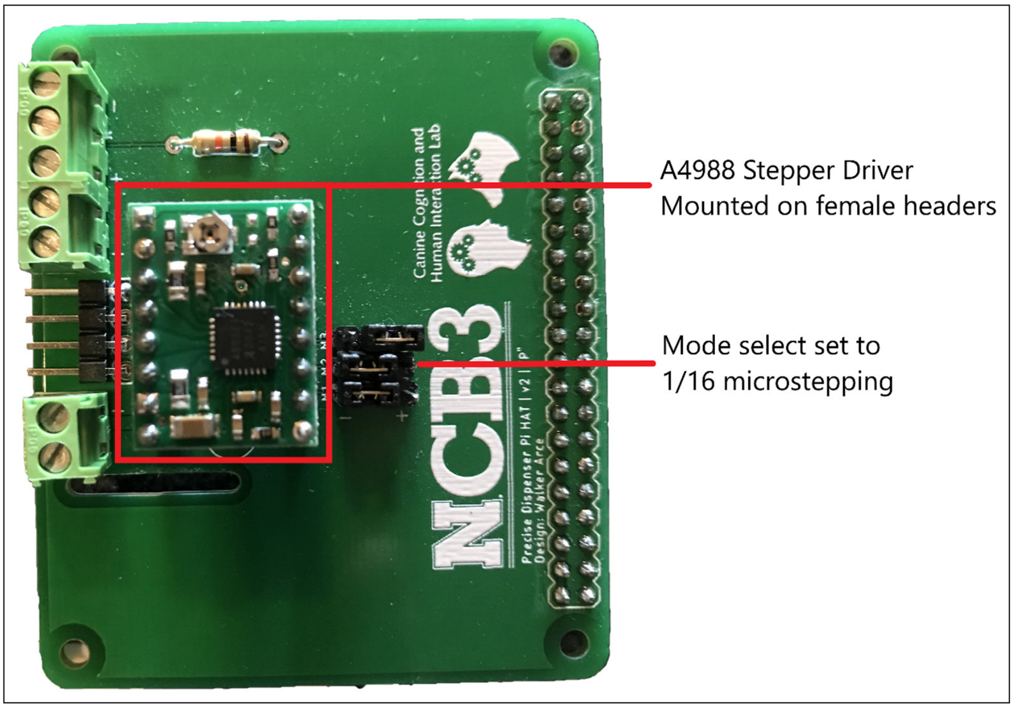


Figure 9 Assembled Pi HAT board.

When fully assembled, the dispenser is eight inches wide by five inches tall (*Figure 10*). The final dispenser has the data connections broken out on the backplate (*Figure 11*). The connections include the four USB ports, HDMI, and USB C power input for the Raspberry Pi, the 12V power supply for the motor, and a global power disconnect. This allows the Raspberry Pi 4 to drive a screen to display stimuli and for an IR touchscreen adapter to be placed on the screen and connected to the Pi USB port as a mouse input device. This device can also be network connected through the Ethernet jack exposed on the backside.

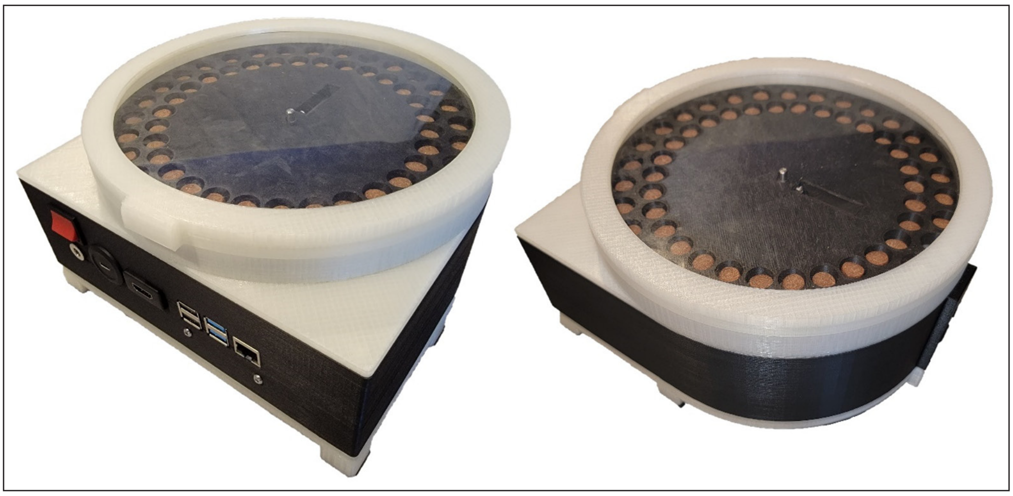


Figure 10 Images of the assembled dispenser, fully loaded with treats.

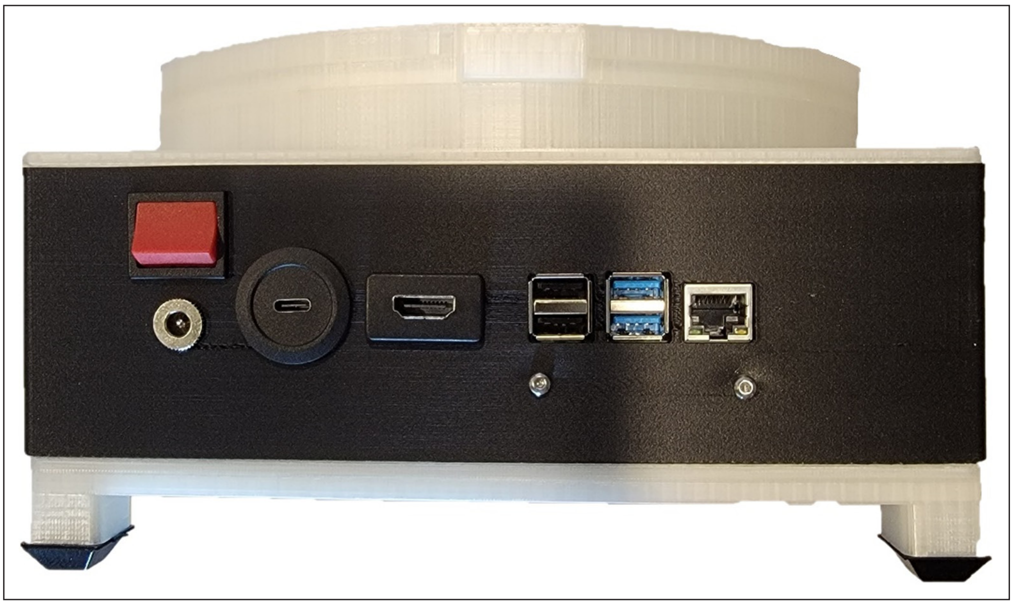


Figure 11 Assembled dispenser connector backplane.

This design can be printed using commodity 3D printers and the dispensers were printed on a Sovol SV01 3D printer. The compact dispenser platform can be assembled in a few hours. The dispenser, when turned on, will create a WiFi access point called, "precise-canine-xx", with "xx" being the dispenser number, i.e., "01" with a local address of `pi@192.168.0.1` and the password `raspberry`, which are the default values in the *pi-ap* project. This can be modified in the *pi-ap* software. By connecting to this, the Raspberry Pi can be accessed through SSH using a program such as PuTTY (Tatham 2020).

Since the Raspberry Pi 4 has an on-board Bluetooth modem, this allows for integration of other custom devices such as inertial measurement units (IMUs), which measure its own movement in terms of acceleration in the X, Y, and Z axes, to allow a canine's position and movement to control treat dispensing (Protopopova et al. 2016; Majikes 2017). Additionally, the sound output of the Pi can allow for sound triggers in the operant conditioning protocol (Protopopova et al. 2016; Edwards 2019).

SOFTWARE DEVELOPMENT

The developed software runs natively on the Raspberry Pi 4 with a Raspberry Pi OS distribution. Python (Python Software Foundation 2019) was used for this development.

The Python class that was developed to control the dispenser controls the stepper motor driver through the GPIO pins and monitors a logic low interrupt from the IR break beam. This class also keeps track of the number of treats left in the jogger and will reject dispensing amounts that exceed the remaining treats. An internal timer is used to track if an exception should be raised, which is controlled by the timeout variable in the `__init__` function.

```
class PreciseDispenser
        self.loaded_treats
                The number of treats remaining the in the jogger
        self.dispensing_timeout
                The amount of time between IR break beam events before an exception
                is thrown
        self.dispensing
                A Boolean to track if the dispenser is currently dispensing
        __init__(loaded_treats, timeout=1)
                Initializer for the canine treat dispenser, configures control
                variables and Pi GPIO.
                :param loaded_treats: The number of treats that are loaded into
                the dispenser for this session. Defaults to max, 59 treats.
                :param timeout: The amount of time before the dispenser throw an
                error between dispensation events
        dispense_treat()
                Function to dispense a single treat. Should not be used, use
                dispense_treats for error handling.
                :returns: True for successful dispensation of a single treat,
                False for a failure to dispense the treat.
        dispense_treats(num_treats)
                Dispenses the specified number of treats. If the requested number
                of treats is greater than the remaining treats, a ValueError is
                thrown. If the treat dispenser loop times out, a ValueError is
                thrown.
                :param num_treats: A number between 1 and the remaining number of
                treats.
                :returns: None. Throws ValueError when an error event is
                encountered.
        treat_dispensed(channel):
                A callback function for when a treat is dispensed. This is
                detected on the falling edge of the IR break beam.
                :param channel: The channel to identify the callback source.
                :returns: None.
        close()
                Closes the dispenser object and cleans up the GPIO assignments.
                :returns: None.
```

The dispenser object can be instantiated with a `PreciseDispenser()` call. This sets the GPIO assignments and starts the needed thread for monitoring the IR break beam. Calling the `dispense_treats(num_treats)` function will initiate the dispenser to dispense the number of treats specified. Putting this into a try-catch and catching for, and printing, ValueError will show the specific error caused during operation.

Each call to dispense treats should be wrapped in a try-except block to catch any errors that bubble up during operation. When the dispenser is no longer being used, it should be closed so the GPIO assignments can be cleaned up and no erroneous behavior is produced. To prevent over-dispensation, there is an internal counter that checks the number of steps for each dispensation. If 54 steps are performed without an IR beam trigger, then it assumes that a treat was dispensed. This compensates for errors in the IR break beam.

The Raspberry Pi image used in this study was exported and a download link was placed in the GitHub repository readme.md file. This image includes a tested PsychoPy (v3.2.4) build and the discussed libraries.

(2) QUALITY CONTROL

SAFETY

Operators of this dispenser should be aware of three hazards:

- Finger pinches.
- Electric shock.
- Burns when assembling electronics.

FINGER PINCHES

The dispenser wheel turns continuously from a stepper motor to dispense treats. This wheel dispenses through an eyelet that, if not careful, can cause the user's finger to be pinched when trying to fix a jam.

ELECTRIC SHOCK

The powering of electronics using an external power source, such as when testing a newly developed circuit, can cause electric shocks if not careful. Worktables should be placed where there is no risk of splashing water. The stepper motor is run using a $12\ V_{DC}$ source. All cables used to connect sensors and power should be inspected for damage and replaced if necessary.

BURNS WHEN ASSEMBLING ELECTRONICS

The assembly of the described electronics includes the use of soldering irons and hot air rework tools. These tools use high temperatures to melt solder and pose a risk of burning when used improperly. Ensure that the user's manuals are read and adhered to when using new tools.

GENERAL TESTING

To prepare the dispenser for operation, treats must be placed in the dispenser. Place treats into all the holes of the treat jogger and manually rotate the wheel to ensure all treats fall out of the holes. If some stick, then make sure any strands are removed from the print. The `dispenser_test.py` script can be used to ensure the break beam works and that treats are dispensed as expected. For the Pet Botanics Training Rewards, ensure that the treats are placed flat.

If the dispenser under dispenses, ensure the treat jogger is mounted securely to the motor shaft. If the set screw strips the hole, use super glue to affix the wheel to the shaft. If the dispenser over dispenses, ensure that the jogger is perfectly aligned over the pellet exhaust before use.

We built five dispensers and tested each one with a similar setup as the retrofitted dispenser (Arce & Stevens 2020). For each dispenser, we performed ten reliability trials for ten treat quantities: 1, 2, 3, ..., 9, and 10 treats. Therefore, each dispenser completed 100 trials with a total of 550 treats. The over- and under-dispensing rates, accuracy, and error rates are shown in *Table 1*. The mean error rate for the five dispensers ranged from 1–9%, with a mean error rate of a 4% chance of over- or under-dispensing a treat. Of those errors, 95% were over-dispensing errors.

	TOTAL TRIALS	CORRECT TRIALS	OVER-DISPENSED TRIALS	UNDER-DISPENSED TRIALS	ACCURACY	ERROR RATE
Dispenser One	100	91	8	1	91%	9%
Dispenser Two	100	93	7	0	93%	7%
Dispenser Three	100	99	1	0	99%	1%
Dispenser Four	100	98	2	0	98%	2%
Dispenser Five	100	99	1	0	99%	1%
Mean		**96**	**3.8**	**0.2**	**96%**	**4%**

Table 1 Results of the dispenser reliability testing.

Considering that the number of treats affects the possibility of errors occurring, we also calculated the accuracy and error rate across all ten treat quantities tested (*Table 2*). Dispensing fewer than eight treats resulted in very low error rates (≤2%). But that rate increased to 6–16% when dispensing eight or more treats. Data are available in the project repository.

	ACCURACY	ERROR RATE
One Treat	100%	0%
Two Treats	100%	0%
Three Treats	100%	0%
Four Treats	100%	0%
Five Treats	100%	0%
Six Treats	98%	2%
Seven Treats	100%	0%
Eight Treats	84%	16%
Nine Treats	94%	6%
Ten Treats	84%	16%

Table 2 Cross quantity accuracy and error metrics.

The first two dispensers showed a higher error rate, which was due to differences in construction on the first two units and to developing the troubleshooting strategy to ensure accurate dispensing.

(3) APPLICATION

USE CASE(S)

A potential use case for this hardware and software would be a canine numerical preference task. In this scenario, the dispenser would be connected to a computer monitor with a touchscreen. The Raspberry Pi could present choices between different numbers of items on the screen (e.g., 3 vs. 6 dots) and the touchscreen could record which of the options the dog selects by touching the monitor with their nose. Then, depending on the choice, the dispenser would dispense either three or six treats. The program could then cycle through a series of pairs of numbers, giving the dog multiple opportunities to choose between different numbers of treats. The assumption is that, assuming dogs can discriminate between the two numbers, they should choose the option with the most items (and therefore treats). This would then allow researchers to record which pairs of numbers dogs have consistent preferences for larger amounts, thereby inferring that dogs can discriminate between those quantities.

This is just one use case for a treat dispenser. As shown in the General Testing section, this device is well-suited for canine research quite generally due to its low error rate. Given its higher precision, it can used in all of the scenarios developed for the retrofitted version (Arce & Stevens 2020) plus situations that require more precision dispensing.

REUSE POTENTIAL AND ADAPTABILITY

With the included Raspberry Pi 4, custom sensors can be easily integrated through Python scripting which can trigger the dispensing command. The developed Python class and custom hardware allow for control of new dispensers in a variety of situations, such as for bird or rat pellets. The integrated Raspberry Pi 4 also allows for machine learning integration using a connected camera, such as for canine gesture recognition (Stock & Cavey 2021).

This dispenser can act as a base for add-ons and upgrades for new forms of canine cognition research. The Raspberry Pi still has a significant number of unused data ports and I/O lines, so new Pi HATs can be developed, and new utilities developed to facilitate this research.

With the onboard WiFi and Bluetooth, toys with embedded sensors can be developed as a trigger input or bark sensors can be developed for reinforcement learning. The ability to decouple the trigger input through wireless communication allows for new interaction modalities to be developed with this platform.

Additionally, the Pet Botanics Training Reward treats are not the only treats that work with the dispenser; treats smaller than 15 mm can be used. If the movement of the treats is not satisfactory with an alternative brand, then the treat jogger can be modified to accommodate the change. Treats larger than 15 mm will most likely be unable to fall through the pellet exhaust, but experimentation may yield different results. In addition, though developed for dogs, the dispenser can be used or adapted to dispense treats for other species.

(4) BUILD DETAILS

AVAILABILITY OF MATERIALS AND METHODS

All parts have been sourced through Amazon, Adafruit, Sparkfun, or DigiKey. These can be readily purchased in single quantities, and many have suitable alternatives. The dispenser itself can be printed using standard fused deposition modeling (FDM) 3D printing methods on personal-grade 3D printers. The 3D printing material is standard polylactic acid (PLA), preferably without dyes.

EASE OF BUILD

The parts used in the build are representative of common parts available from online vendors such as Adafruit and DigiKey. The dispenser requires the use of common hand tools for assembly: hex keys, screwdrivers, and pliers. Additionally, a 3D printer is needed to fabricate the body of the dispenser. Simple soldering skills are needed for assembling the Pi HAT, which consists entirely of through hole components with pitch greater than or equal to 0.1". The PCB can be ordered from JLCPCB by uploading the Gerber files found in the repository (https://github.com/unl-cchil/canine_precise_dispenser/tree/main/hardware/pi_hat/gerbers) (JLCPCB, 2021).

OPERATING SOFTWARE AND PERIPHERALS

For the Raspberry Pi 4, the Raspberry Pi OS was used as well as *RPi.GPIO* (version 0.7.0), *PsychoPy* (version 3.2.4), and *pi-ap* (version 1.10.02).

DEPENDENCIES

This build requires the use of *pi-ap* (Houlahan 2019), Raspberry Pi OS (Raspberry Pi Foundation 2020), and *RPi.GPIO* (version 0.7.0). The driving software to interface with the developed device is written in Python 3.7.6 (Python Software Foundation 2019) and instantiates a basic class to encapsulate the dispenser function.

HARDWARE DOCUMENTATION AND FILES LOCATION

Archive for hardware documentation and build files

Name: GitHub

Persistent identifier: https://github.com/unl-cchil/canine_precise_dispenser

Publisher: Jeffrey R. Stevens

Date published: *2021–10–07*

Hardware Licence: CERN-OHL-S-2.0

Documentation Licence: Creative Commons CC-BY-SA, 4.0

Software is stored together with the hardware files, in the same repository.

Licence: MIT License

Date published: *2021–10–07*

(5) DISCUSSION

CONCLUSIONS

This study presents an all-in-one canine precision treat dispenser that is fully 3D printed. By using the common Raspberry Pi platform, Python-based experiment scripts can be used to control inputs and outputs and new modalities of canine interaction with the experiment can be easily added. This dispenser bridges the functionality and accuracy gap that was demonstrated in the previously developed retrofitted dispenser that we developed (Arce & Stevens 2020). The previous dispenser had an average error rate of 27%, so our presented dispenser achieves much more dependable precision with an error rate of 4%. This allows the dispenser to be used reliably in canine studies requiring precise treat dispensing, such as those focusing on numerical cognition.

By making this work fully open-source and DIY, more researchers can use this dispenser to accelerate and facilitate more complex canine cognition research tasks. Having a single-board computer embedded within the design can allow experimental control using the tested and supported *PsychoPy* Python package (Peirce 2007). Since the dispenser consists of a fully contained experiment platform, it is entirely mobile, meaning that experiments can be conducted outside of the laboratory.

The dispenser requires basic skills in 3D printing, soldering, and use of hand tools to assemble. The most difficult part of the build is the 3D printing, as each piece can up to 24 hours to print. Considering automated reward dispensers are difficult to build from scratch, our proposed dispenser lowers the skills gap significantly, allowing researchers to integrate an automated dispenser into their studies (Hardin et al. 2015; Protopopova et al. 2016; Majikes 2017; Wallis et al. 2017; Robinson et al. 2020). With the rather generic approach of the design, it can be used in any treat or food dispensing task where specific quantities are needed.

The average error rate of the five tested dispensers is 4%, meaning that roughly every 20th dispensing command will result in the wrong number of treats being dispensed. These errors seem to be driven by inaccuracies in the IR phototransistor, where it does not trigger when the treat dispenses. This was compensated for by counting the number of steps so if a treat *should* have been dispensed but the break beam wasn't triggered, a treat is assumed to have been dispensed. This requires trust in the hardware and ensuring that the treat jogger always drops the treat when it's over the pellet exhaust. Treats can be snagged by the jogger because of stranding from the 3D printing process, so cleaning the 3D prints is very important in this case. Additionally, ensuring that the treat jogger is firmly affixed using the set screw is important, as steps can be lost from the shaft slipping.

For studies in which precise dispensing is a necessity, our proposed dispenser meets that need and can be built economically and with easily accessible tools, such as low-cost 3D printers, soldering irons, and hex keys.

FUTURE WORK

Next steps in the work can include the development of a touchscreen being built directly on the dispenser. By using a 7-inch touchscreen display from the Raspberry Pi Foundation, a mount can be developed that connects directly to the dispenser body, creating an all-in-one canine cognition research station that can be transported anywhere. Additionally, a camera can be mounted to the dispenser to evaluate machine learning reinforced canine cognition research (Stock & Cavey 2021).

FUNDING INFORMATION

This research was supported by a National Science Foundation grant (SES-1658837) and a University of Nebraska-Lincoln Research Council Grant-in-Aid to JRS.

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHOR CONTRIBUTIONS

Design, assembly, testing, paper writing and editing, Walker Arce.

Oversight, funding, paper writing and editing, Jeffrey R. Stevens.

AUTHOR AFFILIATIONS

Walker Arce *orcid.org/0000-0003-0819-0710*
Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, US

Jeffrey R. Stevens *orcid.org/0000-0003-2375-1360*
Department of Psychology, Center for Brain, Biology & Behavior, University of Nebraska-Lincoln, US

REFERENCES

Abdullahi, Z., Dost, A., Moore, G., Shrestha, J., Wignall, R., Peixoto, N., & **Neuber, K.** 2019. Wheelchair-mounted dog treat dispenser. Available at: *http://treatdispenser.onmason.com/files/2019/10/492-DESIGN-DOC.pdf*.

Arce, W., & **Stevens, J. R.** 2020. Developing a computer-controlled treat dispenser for canine operant conditioning. *Journal of Open Hardware*, 4(1): 6. DOI: *https://doi.org/10.5334/joh.27*

Autodesk. 2021. Fusion 360. Available at: *https://www.autodesk.com/products/fusion-360/personal*.

Autodesk. 2021. Eagle. Available at: *https://www.autodesk.com/products/eagle/overview*.

Brannon, E. M. 2006. The representation of numerical magnitude. *Current Opinion in Neurobiology*, 16(2): 222–229. DOI: *https://doi.org/10.1016/j.conb.2006.03.002*

Davis, H., & **Perusse, R.** 1988. Numerical competence: From backwater to mainstream of comparative psychology. *Behavioral and Brain Sciences*, 8: 47–68. DOI: *https://doi.org/10.1017/S0140525X00053693*

Edwards, T. L. 2019. Automated canine scent-detection apparatus: Technical description and training outcomes. *Chemical Senses*, 44(7): 449–455. DOI: *https://doi.org/10.1093/chemse/bjz039*

Hardin, D. S., Anderson, W., & **Cattet, J.** 2015. Dogs can be successfully trained to alert to hypoglycemia samples from patients with type 1 diabetes. *Diabetes Therapy*, 6(4): 509–517. DOI: *https://doi.org/10.1007/s13300-015-0135-x*

Houlahan, T. 2019. pi-ap. Available at: *https://github.com/f1linux/pi-ap*.

JLCPCB. 2021. JLCPCB. *https://jlcpcb.com/*.

Majikes, J. J. 2017. Computational and design techniques for a semi-autonomous computerized dog-training system with timing and accuracy performance comparable to a professional dog trainer. Unpublished thesis (PhD), North Carolina State University. Available at: *https://repository.lib.ncsu.edu/bitstream/handle/1840.20/34964/etd.pdf?sequence=1*.

Miklósi, Á. 2015. *Dog Behaviour, Evolution, and Cognition* (Second Edition). Oxford: Oxford University Press.

Peirce, J. W. 2007. PsychoPy—Psychophysics software in Python. *Journal of Neuroscience Methods*, 162(1–2): 8–13. DOI: *https://doi.org/10.3758/s13428-018-01193-y*

PetSafe. 2020. Treat & Train™ Remote Reward Dog Trainer. Available at: *https://intl.petsafe.net/en-gb/support/treat-train-remote-reward-dog-trainer*.

Pololu Robotics and Electronics. 2015. Setting the Current Limit on Pololu Stepper Motor Driver Carriers. Upload 14 April. Available at: *https://www.youtube.com/watch?v=89BHS9hfSUk*.

Protopopova, A., Kisten, D., & **Wynne, C.** 2016. Evaluating a humane alternative to the bark collar: Automated differential reinforcement of not barking in a home-alone setting. *Journal of Applied Behavior Analysis*, 49(4): 735–744. DOI: *https://doi.org/10.1002/jaba.334*

Python Software Foundation. 2019. Python Release Python 3.7.6. *Python.org*. Available at: *https://www.python.org/downloads/release/python-376/*.

Raspberry Pi Foundation. 2020. Teach, Learn, and Make with Raspberry Pi – Raspberry Pi. Available at: *https://www.raspberrypi.org/*.

Robinson, C., Brulé, E., Jackson, J., Torjussen, A., Kybett, J., & **Appshaw, T.** 2020. Tricks and treats: Designing technology to support mobility assistance dogs. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu, HI, USA on 25–30 April 2020, pp. 1–14. DOI: *https://doi.org/10.1145/3313831.3376188*

Stock, J., & **Cavey, T.** 2021. Who's a Good Boy? Reinforcing Canine Behavior in Real-Time using Machine Learning. *arXiv preprint,* arXiv:2101.02380.

Tatham, S. 2020. Download PuTTY: latest release (0.74). Available at: *https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html*.

Thomas, O. M., Fialli, A. S., & **Carlstrom, O. M.** 2021. Design of an Autonomous Canine Entertainment Device. Available at: *https://digital.wpi.edu/concern/student_works/dr26z158d*.

Utrata, E., Virányi, Z., & **Range, F.** 2012. Quantity discrimination in wolves (Canis lupus). *Frontiers in Psychology*, 3, p.505. DOI: *https://doi.org/10.3389/fpsyg.2012.00505*

Wallis, L. J., Range, F., Kubinyi, E., Chapagain, D., Serra, J., & **Huber, L.** 2017. Utilising dog-computer interactions to provide mental stimulation in dogs especially during ageing. In *Proceedings of the Fourth International Conference on Animal-Computer Interaction*, New York, NY, USA on November 2017, pp. 1–12. DOI: *https://doi.org/10.1145/3313831.3376188*

Wayland, M., & **Landgraf, M.** 2018. A Cartesian coordinate robot for dispensing fruit fly food. *Journal of Open Hardware*, 2(1): 3. DOI: *https://doi.org/10.5334/joh.9*

TO CITE THIS ARTICLE:
Arce, W., & Stevens, J. R. 2022. A Precise Dispenser Design for Canine Cognition Research. *Journal of Open Hardware*, 6(1): 2, pp. 1–13. DOI: *https://doi.org/10.5334/joh.41*

Published: 02 February 2022

Journal of Open Hardware is a peer-reviewed open access journal published by Ubiquity Press.