Taylor & Francis
Taylor & Francis Group

Ringing & Migration

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tram20

A Raspberry Pi-based, RFID-equipped birdfeeder for the remote monitoring of wild bird populations

Mason Youngblood

To cite this article: Mason Youngblood (2019) A Raspberry Pi-based, RFID-equipped birdfeeder for the remote monitoring of wild bird populations, Ringing & Migration, 34:1, 25-32, DOI: 10.1080/03078698.2019.1759908

To link to this article: https://doi.org/10.1080/03078698.2019.1759908

Published online: 07 May 2020.

RINGING & MIGRATION
2019, VOL. 34, NO. 1, 25–32
https://doi.org/10.1080/03078698.2019.1759908

A Raspberry Pi-based, RFID-equipped birdfeeder for the remote monitoring of wild bird populations

Mason Youngblood

Psychology, The Graduate Center, City University of New York, New York, NY, USA; Biology, Queens College, City University of New York, Flushing, NY, USA

ABSTRACT
Radio-frequency identification (RFID) is an increasingly popular wireless technology that allows researchers to monitor wild bird populations from fixed locations in the field. I have developed an RFID-equipped birdfeeder based on the Raspberry Pi Zero W, a low-cost single-board computer, that collects continuous visitation data from birds marked with passive integrated transponder (PIT) tags. Each birdfeeder has a perch antenna connected to an RFID-reader board on a Raspberry Pi powered by a portable battery. When a tagged bird lands on the perch to eat from the feeder, its unique code is stored with the date and time on the Raspberry Pi. These birdfeeders require only basic soldering and coding skills to assemble, and can easily be outfitted with additional hardware like video cameras and microphones. I outline the process of assembling the hardware and setting up the operating system for the birdfeeders. Then, I describe an example implementation of the birdfeeders to track House Finches *Haemorhous mexicanus* on the campus of Queens College in New York City.

ARTICLE HISTORY
Received 19 January 2020
Accepted 10 March 2020

Biologging, or the use of small electronic devices to track changes in an animal's physiology, environment, or behaviour, allows researchers to collect large amounts of data passively, without direct observation (Rutz & Hays 2009, Chmura et al 2018). One of the most popular forms of biologging in ornithological research is radio-frequency identification (RFID) (Williams *et al* 2019), a wireless technology that allows researchers to monitor wild bird populations at fixed locations in the field (Bonter & Bridge 2011, Ponchon *et al* 2013). Birds are tagged with leg bands containing a passive integrated transponder (PIT) that transmits a unique code when activated by close proximity to an antenna connected to an RFID reader. The lack of an internal power source means that PIT tags are small enough to have negligible effects on survival (Adelman *et al* 2015). In addition, birds carrying PIT tags do not have to be recaptured and repeatedly handled to yield data, making this method much less invasive than traditional capture–recapture. Based on visitation patterns at RFID-equipped birdfeeders, researchers can collect basic information on movement and foraging ecology (Williams *et al* 2019), and even infer social interactions to reconstruct population-level behavioural patterns such as social networks and dominance hierarchies (Adelman *et al* 2015).

In the past, the high cost of commercial RFID readers has been a financial barrier to their use in research. Previous DIY readers have been significantly cheaper, but typically require a background in circuit building (Bridge & Bonter 2011, Ibarra *et al* 2015, Zárybnická *et al* 2016, Bridge *et al* 2019). I have developed a more user-friendly RFID-equipped birdfeeder (Figure 1), based on the Raspberry Pi Zero W, that requires only basic soldering and coding skills to assemble and activate. Each birdfeeder has a perch antenna connected to an RFID-reader board on a Raspberry Pi powered by a portable battery. When a tagged bird lands on the perch to eat from the feeder, its unique code is stored, with the date and time, on the Raspberry Pi. All the data collected are then backed-up to cloud storage using a personal hotspot Figure 1.

The Raspberry Pi Zero W[1], available for as little as 10 USD, is a single-board computer that is ideal for remote monitoring setups because of its low power consumption and built-in Wi-Fi connectivity. The total cost of the electrical components for this setup, including a battery and antenna, is ~85 USD. Although this is slightly more expensive than the cheapest DIY models, which cost about 30 USD excluding the battery (Bridge & Bonter 2011, Bridge *et al* 2019), it provides more flexibility in data management and can easily be

CONTACT Mason Youngblood myoungblood@gradcenter.cuny.edu
[1]https://www.raspberrypi.org/products/raspberry-pi-zero-w/

Figure 1. One of the RFID-equipped birdfeeders hanging in a tree on the Queens College campus.

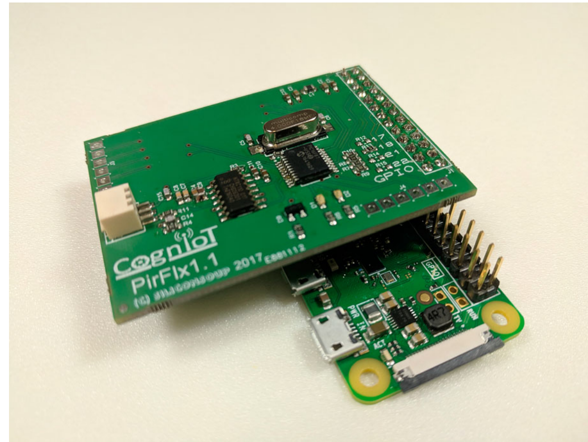

Figure 2. The RFID reader attached to the Raspberry Pi.

outfitted with additional hardware like cameras (Jolles *et al* 2018, Alarcón-Nieto *et al* 2018), microphones (Whytock & Christie 2017, Myers *et al* 2019), and environmental sensors (Sethi *et al* 2018, Grindstaff *et al* 2019). To my knowledge, the only other applications of the Raspberry Pi in RFID-based research have used more expensive models with significantly higher power consumption (Lendvai *et al* 2015, Meniri *et al* 2019).

Below, I outline the process of assembling the hardware and setting up the operating system for the birdfeeders. Then, I describe an example implementation of the birdfeeders to track House Finches *Haemorhous mexicanus* on the campus of Queens College in New York City, and highlight several minor technical issues that we encountered using this system.

Hardware

Raspberry Pi

Each Raspberry Pi will require a microSD card for the operating system and internal storage. I also recommend that you purchase a mini-HDMI to HDMI adaptor and a micro-USB to USB adaptor, to set the system up with a monitor and keyboard. Before the computer can be used with the RFID reader, a standard 40-pin GPIO header[2] needs to be soldered to the board. Alternatively, you can spend slightly more for the Raspberry Pi Zero WH (14 USD) which comes with the GPIO header already installed.

RFID Reader

This system utilises the 125-kHz RFID reader available from CognIoT[3] (46 USD). This reader connects to the GPIO pins next to the microSD port (1–36) with the chip overlapping the body of the computer (Figure 2). The reader requires input from an antenna tuned to 770 microhenries. Please note that the production line for these readers is limited; orders of more than 20 units may therefore require negotiation with the vendor Figure 2.

Power Supply

For power I utilised the Poweradd Pilot X7[4] (26 USD), a power bank with a capacity of 20 ampere hours that can run the setup continuously in the field for four days. These take around 10 hours to charge: thus, for projects that require constant data collection, it may be a good idea to have two batteries for each birdfeeder that can be used alternately.

Tags

I recommend ordering EM4102 PIT tags from Eccel Technology, as they come pre-embedded in leg bands

[2]https://www.adafruit.com/product/2822
[3]https://www.tindie.com/products/CognIoT/125khz-rfid-reader-for-raspberry-pi/
[4]https://amzn.to/2OndvHj

of various sizes[5]. Please ensure that you know the band size required for your species, and have the appropriate permits and ethical approval, before placing an order. I use the 2.6-mm bands for House Finches.

Antennas

Building the perch antennas (Figure 3) is the most technically challenging aspect of this setup. Each antenna consists of enamelled magnet wire[6] 8.25 mm in diameter (30 AWG) wrapped around a 40 x6-mm ferrite rod[7]. Wrapping the wire around the rod is achieved by placing one end of the rod in a power drill, anchoring the wire on the other end of the rod with tape, and slowly rotating the rod with the drill. In order to meet the required inductance of 770 ± 50 microhenries, I have found that making four overlapping wraps 5–15 mm from one end (range 10 mm) works well. I recommend having an inductance meter handy, for troubleshooting[8] Figure 3.

Once wrapping is complete, cover each end with a 6-mm rubber tip[9] and the entire rod in 8-mm polyolefin heat-shrink tubing[10]. Then, the two loose wires can be lightly stripped and soldered to 60-cm leads[11]. These leads should be attached to the three-position 1.5-mm connector[12] by crimping each wire to the 24–26-AWG terminals[13] and inserting them into the leftmost and rightmost positions. The last step is to waterproof the entire antenna using a rubber coating spray (i.e. Plasti Dip).

If you have already completed the software and firmware steps below, then you can test the antenna by plugging it into the white connector on the CognIoT board. If it is correctly tuned then the red light should turn green when you present the antenna with a PIT tag. I have found that this antenna design has the highest range of detection (~2 cm) at the unwrapped end of the antenna. In total, the materials required for 50 antennas (including an inductance meter and rubber coating spray) cost ~105 USD before shipping. All hardware components of the setup can be seen in Figure 3.

Software

The computers in each birdfeeder run a modified Linux operating system with scripts allowing them to collect

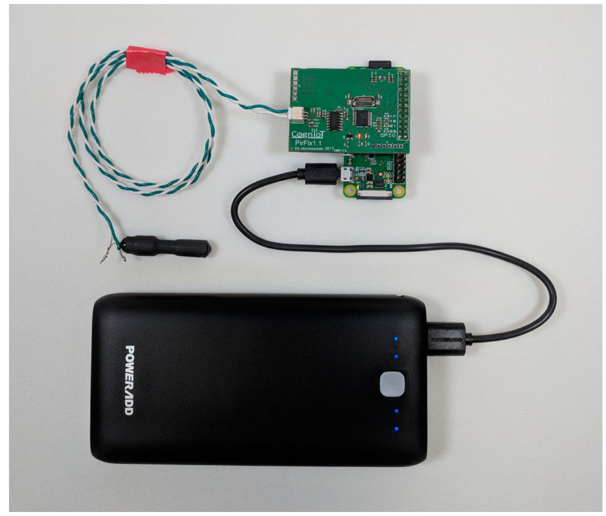

Figure 3. The completed antenna and power supply connected to the computer and RFID reader.

and store RFID data continuously, and upload them to cloud storage after connecting to a personal hotspot generated by a smartphone. These scripts run automatically when the feeder is connected to a power supply, so no commands need to be run in the field. The following steps are needed to set up the operating system.

First, download the most recent version of Raspbian Stretch Lite[14] and follow the instructions to write the image to a microSD card[15]. Once the operating system is installed, boot up the computer and log in using the default username (pi) and password (raspberry). Open the configuration menu:

```
sudo raspi-config
```

Enter 'Network Options' → 'Wi-Fi' to set up the internet connection. Then enter 'Interfacing Options' → 'Serial', and disable the login shell from using the serial port. Make sure that the serial-port hardware is left enabled. Lastly, enable automatic login to the console by entering 'Boot Options'. The computer may require you to reboot before continuing. Next, make sure that the prerequisite software for the RFID reader has been installed:

```
sudo apt-get install python-dev python-setuptools
python-pip
```

[5]https://eccel.co.uk/product-category/avian-products/bird-tags/
[6]http://bit.ly/2AQ8xzc
[7]http://bit.ly/2QmlaGw
[8]https://bit.ly/2wfBq8d
[9]https://amzn.to/2DotDHl
[10]https://amzn.to/2D7Fh8Q
[11]http://bit.ly/2D83z2b
[12]http://bit.ly/2SRmcNj
[13]http://bit.ly/2PFT1ya
[14]https://www.raspberrypi.org/downloads/raspbian/
[15]https://www.raspberrypi.org/documentation/installation/installing-images/

```
sudo apt-get install powertop
sudo pip install wiringpi
sudo pip install numpy
```

Install the example software for the RFID reader:

```
git clone https://github.com/CognIot/RFID_125kHz
```

Import 'rfid_reader.py' and 'schedule.sh' from the data repository into /home/pi, using secure copy or file transfer protocol. The file 'rfid_reader.py' is a modified version of a CognIot script[16] that interfaces with and controls the RFID reader. If you need to change the default polling delay of the RFID reader (262 milliseconds), edit the SetPollingDelay function in 'rfid_reader.py' according to CognIot's documentation. The file 'schedule.sh' stores the collected data into files compatible with *feedr*, an R package for managing and visualising data from RFID-equipped birdfeeders (LaZerte *et al* 2017). Make both files executable:

```
sudo chmod + x rfid_reader.py
sudo chmod + x schedule.sh
```

Remember to modify the 'schedule.sh' script to reflect the name, latitude, and longitude of the site where the birdfeeder will be deployed. To edit the file, you can either open it in the command line with sudo *nano* 'schedule.sh' or use a text editor in the graphical user interface (GUI).

Next, 'schedule.sh' needs to run automatically on start-up, so that a new data file is generated every time the battery is changed. It is also a good idea to run *powertop* to save power. Open the 'rc.local' file located in /etc. Erase the lines for printing the IP address, and add the following lines above 'exit 0' so that they run during start-up:

```
#use powertop auto-tune to save power
sudo powertop –auto-tune &
#run schedule script
/home/pi/schedule.sh &
```

Make sure that 'rc.local' is executable:

```
sudo chmod + x /etc/rc.local
```

Create a folder in /home/pi for the RFID data:

```
mkrdir rfid_logs
```

If you plan to back up the RFID data collected, install *rclone* and use the standard configuration for your cloud-storage service of choice[17]. If you plan on using a cloud-storage service that requires web authentication, then you need to run the configuration via SSH from a computer with a GUI:

```
curl https://rclone.org/install.sh | sudo bash
rclone config
```

There are several ways to handle backups. If the birdfeeders will be in locations with dependable Wi-Fi connections, then you can set up hourly backups via *cron*. Alternatively, you can use software such as *Linux Reader* or *extFS* to copy the files directly from the SD card of the Raspberry Pi. For my situation, I decided to have the birdfeeder automatically upload the RFID data to cloud storage whenever it connects to a personal hotspot generated by a smartphone. The first step is to install and configure an appropriate network manager:

```
sudo apt-get install wicd wicd-curses
sudo systemctl stop dhcpcd
sudo systemctl disable dhcpcd
sudo systemctl start wicd.service
sudo systemctl enable wicd.service
sudo gpasswd -a pi netdev
```

Next, open up the text-based interface for the manager to prioritise and automatically connect to the personal hotspot, and disable automatic connection to any networks that might interfere (e.g. institutional connections requiring login):

```
wicd-curses
```

Import 'drive_backup.sh' from the data repository into /etc/wicd/scripts/postconnect using secure copy or file transfer protocol, remove the '.sh' file extension, and make it executable:

```
sudo chmod + x /etc/wicd/scripts/postconnect/
drive_backup
```

The script 'drive_backup' uploads everything in /home/pi/rfid_logs to /rfid_logs/site_name, a directory that you need to create in your cloud storage. Remember to change 'site_name' to the name of the site where the birdfeeder will be deployed.

Lastly, before the RFID reader can be used with the computer it must be set up to read EM4102 tags. Until this step is completed the reader will not register correctly constructed antennas. Run the setup script and follow the relevant menu options:

```
python RFID_125kHz/python/RFIDReader.py
```

Example

In the spring and summer of 2018, we banded 138 House Finches with PIT tags (92.0% immatures, 3.6% adult females and 4.4% adult males). Due to a high level of

[16]https://github.com/CognIot/RFID_125kHz
[17]https://rclone.org/docs/

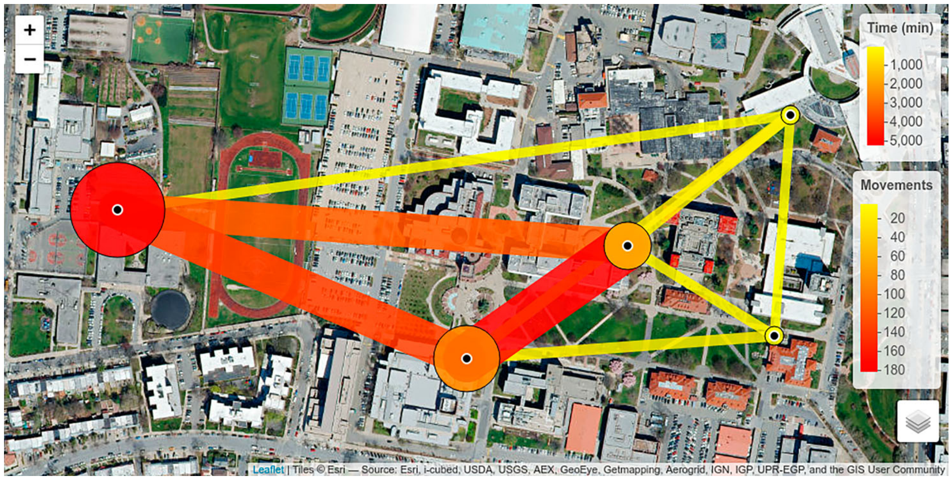

Figure 4. The locations of the five birdfeeders on the Queens College campus, the total time that birds spent at each location (size and colour of each point), and the total number of occasions that birds moved between locations (thickness and colour of each line). Plotted using *feedr* (LaZerte *et al* 2017).

onward dispersal (~91.3% based on return rate), we banded an additional 47 House Finches in the spring and summer of 2019 (63.8% immatures, 8.5% adult females and 27.7% adult males). All birds were captured using perch traps[18]. Immature birds were distinguished from adult females, which also lack the bright colours of adult males, by the colour of their secondary covert feathers (Hill 2002). The protocol for capture and banding was approved by the Institutional Animal Care and Use Committee of Queens College (Protocol #179), and conducted with a federal permit from the United States Department of the Interior (Permit #23708).

Five RFID-equipped birdfeeders were deployed on the Queens College campus from 31 October 2018 to 1 August 2019. Every four days, we visited each birdfeeder to replace the food and batteries and reboot them with a personal hotspot in range. Over the course of the field season we recorded 6878 visits from 28 individual birds (12 banded in 2018 and 16 banded in 2019), after accounting for sequential reads within visits. The total time that individuals spent at each of the five feeders, as well as the number of movements between feeders, can be seen in Figure 4. Birds began to visit the feeders regularly towards the end of March, and activity peaked in early May (Figure 5a). Foraging activity was relatively constant throughout the day, with the highest levels occurring around dusk (Figure 5b), a result consistent with the findings of Bonter *et al* (2013) Figures 4–5.

Once the raw visitation data has been imported into R using the package *feedr* (LaZerte *et al* 2017), it can be used for several different analyses. Here I demonstrate two examples: first, the estimation of the social network of the population and, second, the estimation of the dominance hierarchy of the population.

For species such as House Finches that congregate around birdfeeders, co-occurrences of individuals during foraging bouts can be used to estimate the structure of the social network of a population (Adelman *et al* 2015). To achieve this, adjacent feeder visits by different birds are transformed into association data using machine learning with a Gaussian mixture model (Psorakis *et al* 2015), implemented in the R package *asnipe* (Farine 2013). These association data are then used to reconstruct a weighted social network of the population, where nodes are individual birds and links represent interaction rates between them. The estimated social network of the population can be seen in Figure 6a. In this case, the 25 birds represent a subsample of the entire population. Although partial networks often closely reflect the structure of full networks (Silk *et al* 2015), any social network constructed from a subsample should be interpreted with caution Figure 6.

House Finches have linear dominance hierarchies in which females are typically dominant to males (Thompson 1960a, Belthoff & Gauthreaux 1991), and aggressive interactions between individuals often occur at food sources when one individual displaces another (Thompson 1960b). Accordingly, investigating dominance using displacement data is a well-established method in House Finches (McGraw & Hill 2002, Moyers *et al* 2018), as in other species (Miller

[18]http://www.thirdwheel.biz/perch-traps.html

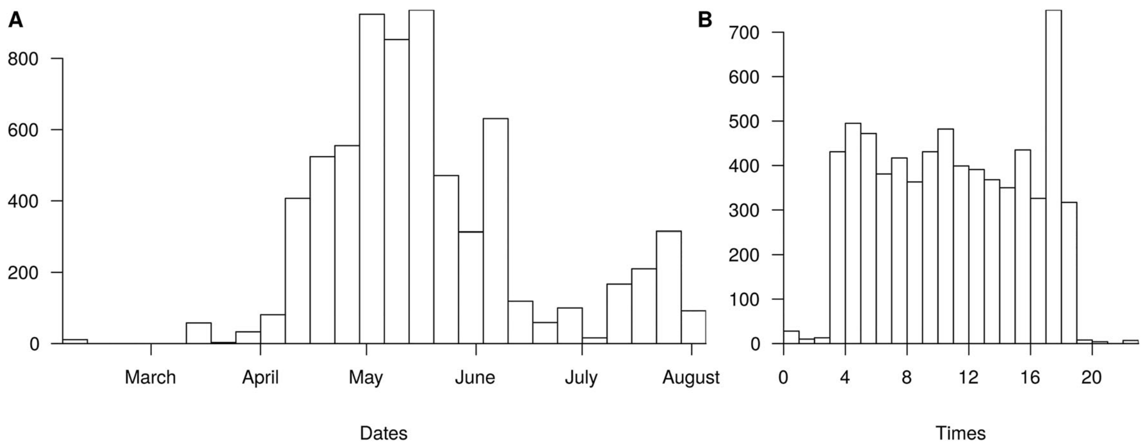

Figure 5. (a) The number of feeder visits (y-axis) that occurred on each day of the study period, plotted by week. (b) The total number of feeder visits (y-axis) that occurred during each hour across the entire study period.

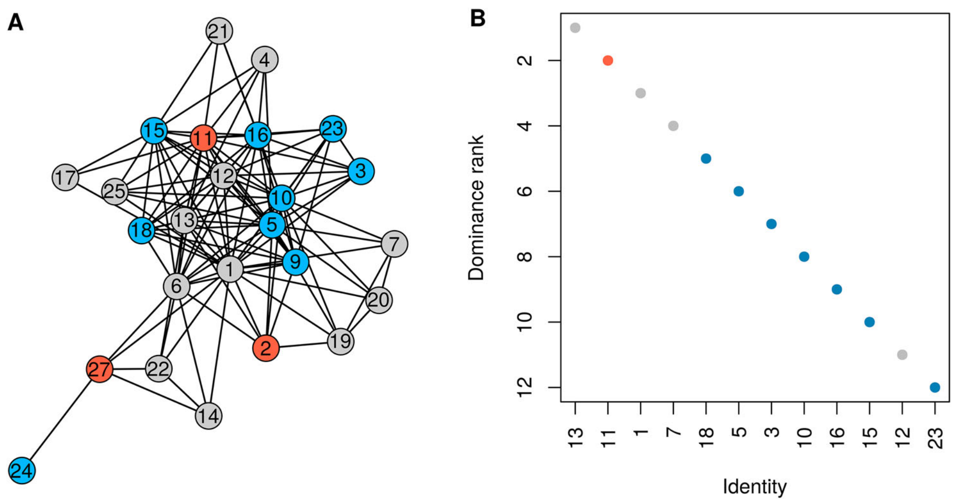

Figure 6. (a) The estimated social network of the 25 birds for which association data were available. (b) The dominance ranks (y-axis) of the 12 birds for which displacement interactions were available. Plotted with *aniDom* (Farine & Sánchez-Tójar 2019). Individual birds are numbered and colour-coded: adult females red, adult males blue, immatures grey.

et al 2017, Evans *et al* 2018). Firstly, displacement events, in which one bird displaces another at the feeder within two seconds, are extracted from the raw visitation data. These pairwise interaction data are then analysed using the *aniDom* package in R (Farine & Sánchez-Tójar 2019). Birds that consistently displace other individuals at the feeders are coded as dominant to those individuals. The estimated dominance hierarchy of the population can be seen in Figure 6b.

Technical limitations

During the field season we periodically experienced three minor technical problems. Firstly, cold weather or high humidity can cause the batteries to shut down prematurely. I highly recommend storing the components in a waterproof box, with packets of silica gel to avoid moisture build-up. If temperatures are approaching freezing, or rain is expected, it is also a good idea to check the batteries daily rather than every four days. Next, sometimes the configuration file for *rclone* is cleared when the backup script runs without the personal hotspot. If files are not appearing in the cloud backup, a quick reconfiguration of *rclone* should solve the issue. Lastly, if the computers reboot without connecting to a personal hotspot the system time will not be accurate. Ensure that the hotspot is on and within range whenever the computers are rebooted. For connection confirmation, I recommend creating a blank text file called 'test' in /home/pi/rfid_logs on each unit. If you delete this file from the online backup and it reappears upon reboot then the computer has connected successfully to the hotspot. Alternatively, you can use a smartphone app that allows you to view which devices are currently connected to your hotspot.

Acknowledgements

I would like to thank Alec Lindsay and Andrew Richards for their invaluable help with antenna design, as well as Shari Zimmerman and Andrea Lopez for their assistance with birdfeeder maintenance and data collection. I would also like to thank David Lahti, as well as all members of the Lahti lab, for their valuable conceptual and analytical feedback.

Data availability statement: all required scripts are available in the Harvard Dataverse repository at https://doi.org/10.7910/DVN/XAIRNM.

Funding

Funding statement: this work was funded by a Research Enhancement Award from Queens College, and a Doctoral Student Research Grant from the Graduate Center, CUNY. Disclosure statement: the author has declared that no competing interests exist.

References

Adelman, J.S., Moyers, S.C., Farine, D.R. & Hawley, D.M. (2015) Feeder use predicts both acquisition and transmission of a contagious pathogen in a North American songbird. *Proceedings of the Royal Society B: Biological Sciences* **282**, 20151429.

Alarcón-Nieto, G., Graving, J.M., Klarevas-Irby, J.A., Maldonado-Chaparro, A.A., Mueller, I. & Farine, D.R. (2018) An automated barcode tracking system for behavioural studies in birds. *Methods in Ecology and Evolution* **9**, 1536–1547.

Belthoff, J.R. & Gauthreaux, S.A.J. (1991) Aggression and dominance in house finches. *Condor* **93**, 1010–1013.

Bonter, D.N. & Bridge, E.S. (2011) Applications of radio frequency identification (RFID) in ornithological research: a review. *Journal of Field Ornithology* **82**, 1–10.

Bonter, D.N., Zuckerberg, B., Sedgwick, C.W. & Hochachka, W.M. (2013) Daily foraging patterns in free-living birds: exploring the predation–starvation trade-off. *Proceedings of the Royal Society B: Biological Sciences* **280**, 20123087.

Bridge, E.S. & Bonter, D.N. (2011) A low-cost radio frequency identification device for ornithological research. *Journal of Field Ornithology* **82**, 52–59.

Bridge, E.S., Wilhelm, J., Pandit, M.M., Morena, A., Curry, C.M., Pearson, T.D., Proppe, D.S., Holwerda, C., Eadie, J.M., Stair, T.F., Olson, A.C., Lyon, B.E., Branch, C.L., Pitera, A.M., Kozlovsky, D., Sonnenberg, B.R., Pravosudov, V.V. & Ruyle, J.E. (2019) An Arduino-based RFID platform for animal research. *Frontiers in Ecology and Evolution* **7**, 257.

Chmura, H.E., Glass, T.W. & Williams, C.T. (2018) Biologging physiological and ecological responses to climatic variation: new tools for the climate change era. *Frontiers in Ecology and Evolution* **6**, 92. doi: 10.3389/fevo.2018.00092

Evans, J.C., Devost, I., Jones, T.B. & Morand-Ferron, J. (2018) Inferring dominance interactions from automatically recorded temporal data. *Ethology* **124**, 188–195.

Farine, D.R. (2013) Animal Social Network Inference and Permutations for Ecologists in R using *asnipe*. *Methods in Ecology and Evolution* **4**, 1187–1194.

Farine, D.R. & Sánchez-Tójar, A. (2019) aniDom: *inferring dominance hierarchies and estimating uncertainty*. R package version 0.1.4. https://CRAN.R-project.org/package=aniDom

Grindstaff, B., Mabry, M.E., Blischak, P.D., Quinn, M. & Chris Pires, J. (2019) Affordable remote monitoring of plant growth in facilities using Raspberry Pi computers. *Applications in Plant Sciences* **7** (8), e11280.

Hill, G.E. (2002) *A Red Bird in a Brown Bag*. Oxford University Press, New York City.

Ibarra, V., Araya-Salas, M., Tang, Y., Park, C., Hyde, A., Wright, T.F. & Tang, W. (2015) An RFID based smart feeder for hummingbirds. *Sensors* **15**, 31751–31761.

Jolles, J.W., Laskowski, K.L., Boogert, N.J. & Manica, A. (2018) Repeatable group differences in the collective behaviour of stickleback shoals across ecological contexts. *Proceedings of the Royal Society B: Biological Sciences* **285**, 13–16.

LaZerte, S.E., Reudink, M.W., Otter, K.A., Kusack, J., Bailey, J.M., Woolverton, A., Paetkau, M., de Jong, A. & Hill, D.J. (2017) *feedr* and animalnexus.ca: a paired R package and friendly web application for transforming and visualizing animal movement data from static stations. *Ecology and Evolution* **7**, 7884–7896.

Lendvai, Á.Z., Akçay, Ç., Weiss, T., Haussmann, M.F., Moore, I.T. & Bonier, F. (2015) Low cost audiovisual playback and recording triggered by radio frequency identification using Raspberry Pi. *PeerJ – Life & Environment* **3**, e877. doi: 10.7717/peerj.877

McGraw, K.J. & Hill, G.E. (2002) Testing reversed sexual dominance from an ontogenetic perspective: juvenile female house finches are dominant to juvenile males. *Ibis* **144**, 139–142.

Meniri, M., Farley, A., Helfenstein, F. & Fasel, N. (2019) A guide for ecologists to build a low-cost selective trap using radio frequency identification detection. *Behavioral Ecology and Sociobiology* **73**, 80.

Miller, E.T., Bonter, D.N., Eldermire, C., Freeman, B.G., Greig, E.I., Harmon, L.J., Lisle, C. & Hochachka, W.M. (2017) Fighting over food unites the birds of North America in a continental dominance hierarchy. *Behavioral Ecology* **28**, 1454–1463.

Moyers, S.C., Adelman, J.S., Farine, D.R., Moore, I.T. & Hawley, D.M. (2018) Exploratory behavior is linked to stress physiology and social network centrality in free-living house finches. *Hormones and Behavior* **102**, 105–113.

Myers, D., Berg, H. & Maneas, G. (2019) Comparing the soundscapes of organic and conventional olive groves: a potential method for bird diversity monitoring. *Ecological Indicators* **103**, 642–649.

Ponchon, A., Grémillet, D., Doligez, B., Chambert, T., Tveraa, T., González-Solís, J. & Boulinier, T. (2013) Tracking prospecting movements involved in breeding habitat selection: insights, pitfalls and perspectives. *Methods in Ecology and Evolution* **4**, 143–150.

Psorakis, I., Voelkl, B., Garroway, C.J., Radersma, R., Aplin, L.M., Crates, R.A., Culina, A., Farine, D.R., Firth, J.A., Hinde, C.A., Kidd, L.R., Milligan, N.D., Roberts, S.J., Verhelst, B. & Sheldon, B.C. (2015) Inferring social structure from temporal data. *Behavioral Ecology and Sociobiology* **69**, 857–866.

Rutz, C. & Hays, G.C. (2009) New frontiers in biologging science. *Biology Letters* **5**, 289–292.

Sethi, S.S., Ewers, R.M., Jones, N.S., Orme, C.D.L. & Picinali, L. (2018) Robust, real-time and autonomous monitoring of ecosystems with an open, low-cost, networked device. *Methods in Ecology and Evolution* **9**, 2383–2387.

Silk, M.J., Jackson, A.L., Croft, D.P., Colhoun, K. & Bearhop, S. (2015) The consequences of unidentifiable individuals for the analysis of an animal social network. *Animal Behaviour* **104**, 1–11.

Thompson, W.L. (1960a) Agonistic behavior in the house finch: annual cycle and display patterns. *Condor* **62**, 245–271.

Thompson, W.L. (1960b) Agonistic behavior in the house finch: factors in aggressiveness and sociality. *Condor* **62**, 378–402.

Whytock, R.C. & Christie, J. (2017) Solo: an open source, customizable and inexpensive audio recorder for bioacoustic research. *Methods in Ecology and Evolution* **8**, 308–312.

Williams, H.J., Taylor, L.A., Benhamou, S., Bijleveld, A.I., Clay, T.A., de Grissac, S., Demšar, U., English, H.M., Franconi, N., Gómez-Laich, A., Griffiths, R.C., Kay, W.P., Morales, J.M., Potts, J.R., Rogerson, K.F., Rutz, C., Spelt, A., Trevail, A.M., Wilson, R.P. & Börger, L. (2019) Optimizing the use of biologgers for movement ecology research. *Journal of Animal Ecology* **89**, 186–206.

Zárybnická, M., Kubizňák, P., Šindelář, J. & Hlaváč, V. (2016) Smart nest box: a tool and methodology for monitoring of cavity-dwelling animals. *Methods in Ecology and Evolution* **7**, 483–492.