blues wireless

Agricultural Monitoring with IoT and Machine Learning

"IoT and TinyML enable low-latency, low power and low bandwidth machine learning at the edge."

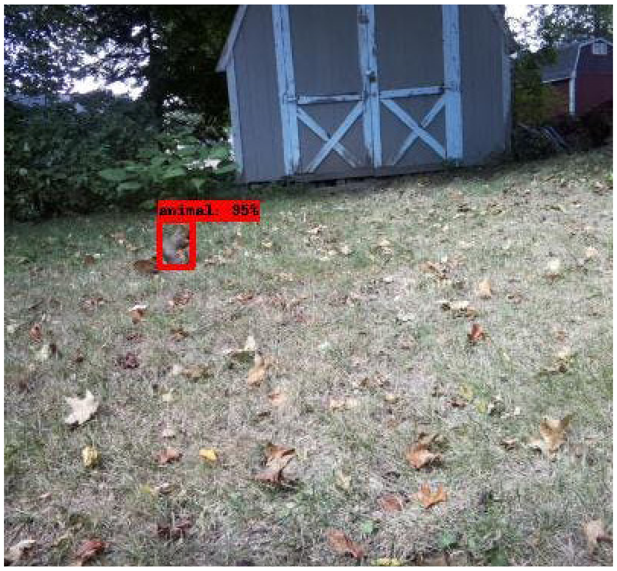
TJ VanToll
Principal Developer Advocate

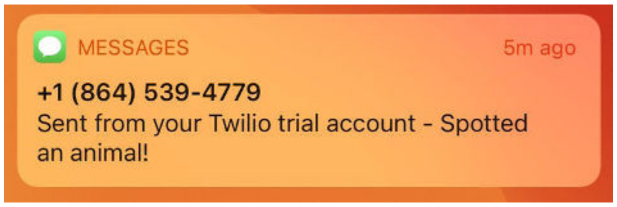
Instructions on GitHub:https://github.com/tjvantoll/pest-detector

IoT and machine learning devices enable growers in remote areas to detect pests and diseases in plants. By taking a picture of the plant and running machine learning models on the device, farmers can correct environmental conditions or apply treatment in real time. If your device runs on cellular, there is no need for an internet connection for the device to work properly - a practical requirement for most farmers.

When building proof-of-concept or prototype IoT devices it is important to spend most of your time on features that solve business problems, not utility functionality. To that end, Blues Wireless Notecard is the simplest, and most cost-effective way to add connectivity to IoT devices. Simply plug the Notecard into an M.2 connector and it'll connect your device to the cellular network automatically, ready to transmit and receive data.

Learn how to build a cellular IoT sensor prototype for continuous monitoring of environments and machines for less than $180, using only 5 hardware components.




MESSAGES 5m ago
+1 (864) 539-4779
Sent from your Twilio trial account - Spotted an animal!

Why TinyML For Sustainable Agriculture

In a world of increasing temperatures and population count, optimizing growing operations quickly becomes a necessity. The global population is expected to reach nearly 10 billion by 2050, requiring a radical shift in the way we produce (and consume) food. IoT and TinyML enable low-latency, low power and low bandwidth machine learning at the edge to deliver actionable intelligence to small-scale and remote farmers that would otherwise not have access.

Handling machine learning at the edge means you don't have to ship image or video files to a cloud software program. Not only will you save megabytes, but you'll also have a quicker, more responsive system. For workloads where privacy is a concern, this means potentially sensitive images don't leave the local device, nor are they stored anywhere.

Using a cloud-based dashboard allows data access from anywhere. For connectivity, use the Blues Wireless Notecard, a cellular and GPS-enabled device-to-cloud data pump that comes with 500 MB of data usable over 10 years. For the fastest development time, plug the Notecard into a Notecarrier, a host board with extensions for headers, battery connections and antennae. Use the Notecarrier Pi for this project because it has headers that are compatible with Raspberry Pi.

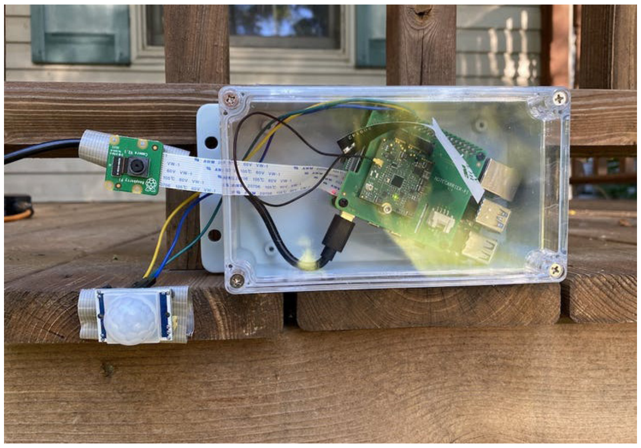
Behind the IoT Agricultural Monitor

If you're looking to start writing and validating a machine learning algorithm on a cellular IoT device prototype in less than a day, this is the best project to follow. You can find the complete source code for the project at the GitHub repository linked below and complete project assembly instructions on Hackster.

GitHub: https://github.com/tjvantoll/pest-detector

Hackster: https://www.hackster.io/tjvantoll/what-s-destroying-my-yard-pest-detection-with-raspberry-pi-890c3a

Price	$179.94	Languages	Python
Lines Of Code	13		
Project Time	5 Hours		



Hardware

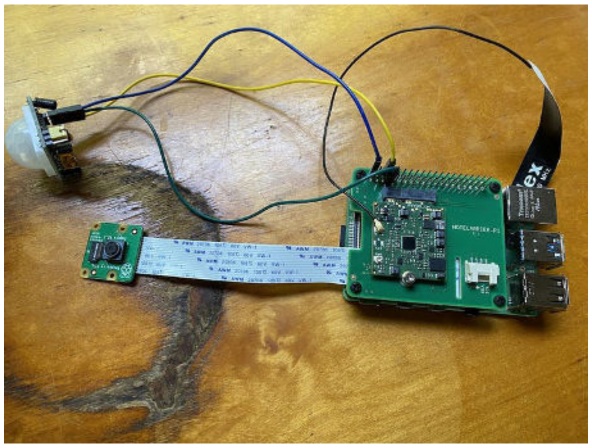
- Blues Wireless Raspberry Pi Starter Kit:
    - Blues Wireless Notecard SoM
    - External cellular antenna
    - Blues Wireless Notecarrier Pi Hat
- Raspberry Pi 4 Model B
- Raspberry Pi Camera Module V2
- PIR Motion Sensor (generic)
- Awclub ABS Plastic Junction Box

Software apps and online services

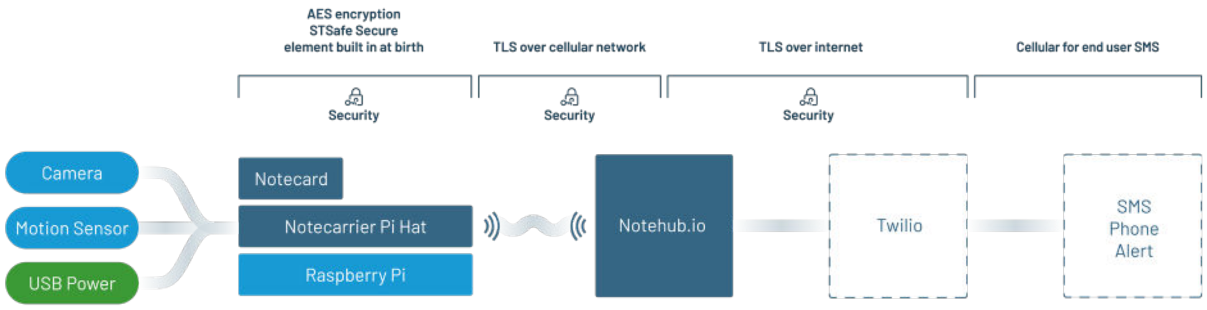
- Blues Wireless Notehub.io
- Twilio SMS Messaging API
- Edge Impulse (custom image classification modeling)
- Microsoft CameraTraps

The main parts of the project are:

- Connecting a microcontroller to a motion sensor and a camera
- Taking images
- Processing images via a machine learning model
- Triggering alerts if the machine learning system detects a targeted pattern



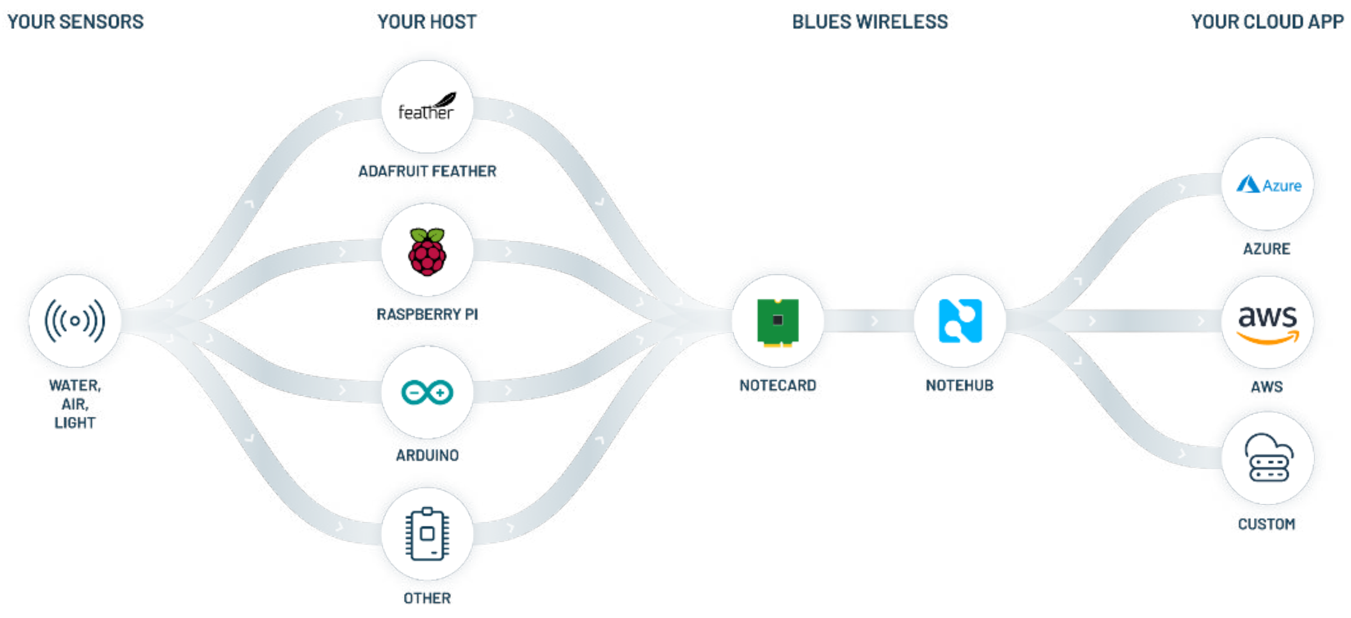
We found it shockingly easy to set up the hardware and software to begin training a model. Your success will be determined by the path you choose to get an appropriate machine learning model. Using Notecard lets you skip past the communications part of your project because the infrastructure, permissions, security, and provisioning all come out of the box for you.



How Blues Wireless Improves Agricultural Monitoring With ML

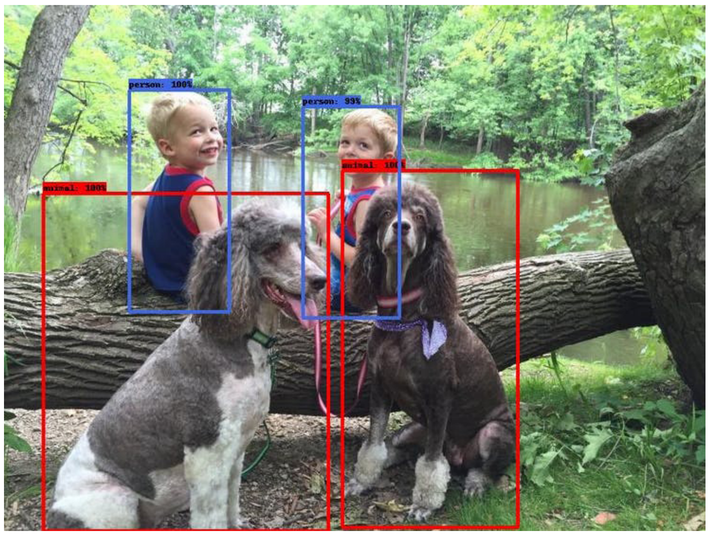
Blues Wireless is the most cost effective and easiest way to add connectivity to a device. In 30 minutes, you can go from unboxing to sending arbitrary data over a global cellular network, with no configuration needed. Building the right Agricultural Monitoring device with Machine Learning is complex enough without having to deal with building a connectivity layer so we advise customers to skip that and use one of our pre-built System on a Module Notecards with zero-configuration provisioning for connectivity.

In the image below, you'll see a left-to-right depiction of how sensor data moves from an edge device to a cloud application. Blues Wireless provides the infrastructure for bidirectional communication between edge devices and cloud endpoints via a combination of hardware and software. On the hardware side, in the host device, Blues Wireless Notecard System on a Module provides an internal endpoint for sensor data. Notecard securely transmits the sensor data to the customer's preferred cloud endpoint via Blues Wireless Notehub, an intermediary cloud application. Notehub provides protocol translation, transport security, data routing, device management, and device firmware update capability.



How Image Classification Works in Machine Learning

The image classification process is built on modeling or training a system to recognize patterns. Once the machine learning model is "trained", the system will produce a file containing the descriptors of the model the Machine Learning software will consume to learn the model. This project required common animals and backyard pests to be identified, so Microsoft's CameraTraps model, MegaDetector, was used.



CameraTraps' database is so extensive, it's too processor-intensive for a Raspberry Pi at a low enough latency. While the MegaDetector model worked for our demonstration project, we recommend using either a prebuilt model with TensorFlow or a custom model with Edge Impulse, because they use more appropriate resources for an IoT project. Be sure to read our documentation on using Edge Impulse with Blues Wireless.

For technical support with the Blues Wireless Notecard and Notecarrier, please visit dev.blues.io.

Other ML and TinyML Applications of This Project

This device is useful for object recognition and alerting. You can extend or repurpose this device by simply changing the ML model. Because Notecard provides bidirectional communication, you can easily send new models from the cloud to your device. A much easier upgrade than going out in the field to manually update devices, isn't it? Remember, devices built using the Blues Wireless Notecard can be used anywhere, even if Wi-Fi access isn't available. Here are some other ideas for this device:

- Speech activation
- Wildfire detection
- Flooding and water leak detection
- Wearable health devices
- Anomaly detection for predictive maintenance
- Intelligent shipping
- Gesture recognition for the visually impaired
- Inventory management
- Smart city devices (people counting, traffic monitoring)
- Wildlife tracking

Follow these instructions to learn how to build an IoT machine learning prototype.