Ray: Batteryless Occupancy Monitoring using Reflected Ambient Light

Anonymous authors
Paper under double-blind review

Abstract

Room-level occupancy-tracking systems enable intelligent control of building functions like air conditioning and power delivery to adapt to the needs of their occupants. Unfortunately, existing occupancy-tracking systems are bulky, have short battery lifetimes, are not privacy-preserving, or only provide course-grain occupancy information. Furthermore, retrofitting existing infrastructures with wired sensors is prohibitively expensive.

In this paper, we present Ray, a *batteryless*, doorframe/passageway-mounted room-level occupancy monitoring sensor that uses changes in indoor ambient light reflections to detect people entering and exiting a room or hallway and estimate direction of travel. We evaluated Ray in mixed lighting conditions on both sides of the doorway in an office-style setting, using subjects with a wide variety of physical characteristics. We conducted 881 controlled experiments in 7 doorways with 9 individuals and achieved a total detection accuracy of 100% and movement direction accuracy averaging 96.4%. Furthermore, we deployed Ray sensors for 64 days in 5 locations, comparing them with a commercial batteryless occupancy sensor. Ray outperformed the commercial sensor, particularly where traffic is moderate to heavy. Ray demonstrates that ambient light reflections provide both a promising low-cost, long-term sustainable option for monitoring how people use buildings and an exciting new research direction for *batteryless* computing.

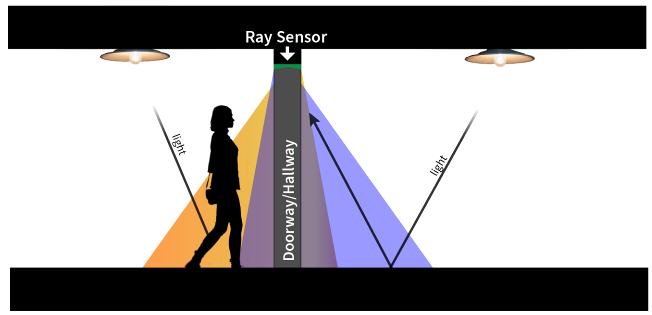


Figure 1: The overall system concept of Ray, a batteryless, energy-harvesting, doorway mounted occupancy tracking and person detection enabling system. This system uses reflective indoor lighting to both power the system and detect person entry and exit activity to a room or corridor.

1 Introduction

Understanding how people move, work, and live within a workplace or residence is essential for enabling health, efficiency, and security applications in smart buildings. Appliances, computers, lighting, heating and cooling systems can adapt their behavior depending on the number of occupants, their needs, and the context of their interactions. Smart buildings can automatically identify indoor traffic patterns, poorly-used space, and congested walkways, helping us better understand how people interact with the indoor spaces they use. We can only achieve these benefits if we can effectively sense how people move indoors.

Unfortunately, current occupancy-tracking systems are large, expensive, and high-maintenance—too expensive for large-scale deployments and too high-maintenance for long-term use. Existing systems use a variety of techniques, including ultrasound [20], images [35, 36], wearables [12], instrumented objects [4], structural vibrations [30], and opportunistic data leaked from existing meters and security systems [40]. Some gather identifiable information. Others require building remodeling, force users to change their behavior, or require structural models of the building. For any of these solutions to work, we must either provide wired power to the sensors (which is usually both expensive and invasive), or use batteries which increase cost, environmental impact, and fire risk, and which must be replaced every few years (even rechargeables).

In this paper we present Ray (overview shown in Figure 1), an occupancy-monitoring sensor that is low-cost and low-maintenance, preserves occupant privacy, and can operate for decades[1] without wired power or batteries.

Like previous solutions [20], Ray attaches to the top of a door frame and monitors movement in and out of the doorway. In contrast, however, Ray does not use active sensors (like ultrasonic range finders), but instead senses movement using the same ambient light reflections that power the sensor. Ray harvests solar energy from indoor lights to power all operations, and uses a combination of hardware and software techniques to detect human movement and direction as solar

[1] Actual lifetimes depend on environmental conditions, enclosure quality, and rates of decay for silicon and other circuit materials. Without the usual bottleneck (the battery), lifetimes of 10–50 years are realistic but not guaranteed.

energy availability changes. Ray stores this information on device, and opportunistically transmits occupancy information to a basestation using its radio.

Contributions:

1. We present a novel system design for unobtrusive, long-term, low-cost, zero-maintenance occupancy tracking.
2. We explore design considerations for batteryless, intermittently-powered sensing systems for detecting ephemeral events that can be broadly applied to other batteryless sensing applications.
3. We provide an implementation, deployment (both controlled and in-the-wild), and evaluation of Ray that explores the strengths and limitations of our approach.

Ray is, to our knowledge, the first batteryless occupancy-monitoring solution [13, 14], and demonstrates the potential and usefulness of long-lived, energy-harvesting, batteryless sensing operation in the built environment. In this paper we present our design, a working prototype, and evaluation results showing the efficacy of the approach.

2 Batteryless People Sensing

Energy-harvesting batteryless sensors are critical to an affordable and sustainable Internet-of-Things (IoT) and the future of smart buildings. Running wires to power new sensors and other devices is expensive and not always feasible. On the other hand, batteries are expensive, bulky, and often hazardous. Even rechargeable batteries wear out after a few years, and replacing trillions of additional batteries every year would be both expensive and irresponsible. In contrast, batteryless sensors powered entirely with harvested energy cost less, weigh less, and can operate for decades with minimal maintenance and environmental impact.

However, batteryless sensing is challenging. Energy is stored in one or more small, cheap capacitors to improve efficiency and responsiveness [17]. Harvested energy is variable and difficult to predict. Power failures are common, interrupting computation and data processing, sensing, and communication. Clocks reset and volatile memory is lost frequently, complicating a developer's ability to build robust and sophisticated applications.

Recent advances in checkpointing [1, 31], consistent execution [6, 25], timekeeping [19], energy management [17], testing [16], and debugging [8] address key challenges and have enabled new and interesting applications. Examples of such applications include tracking building and appliance energy consumption [5, 10] and monitoring greenhouses [17].

In spite of these improvements, current batteryless sensing applications are limited and typically fall into one of two categories: those that depend on an RFID reader and those that opportunistically detect valid, useful data whenever measured. Power failures and long outages make it difficult or impossible to gather streams of uninterrupted data, inevitably resulting in an inferior quality performance when compared to reliably powered sensors. This has complicated the design and deployment of such batteryless sensors in many application areas.

Occupancy-monitoring applications try to instrument buildings, people, or other indoor elements to get a better understanding of the number of people in a room. This information is the baseline data for successful operation of smart building functions such as intelligent temperature and HVAC control, efficiency monitoring, elderly tracking, and other applications. Existing occupancy-monitoring systems use many sensing techniques and deploy in many different form factors, with doorway-based sensing being one promising method [20, 22]. In this paper, we implement a doorway-mounted batteryless sensor for occupancy monitoring and investigate the challenges posed by an unreliable power supply to achieving a reasonable quality of sensing. We recognize three major aspects to implement a successful sensing system with unreliable power:

Intermittence: Small energy storage combined with unpredictable energy harvesting means that batteryless devices must be equipped to handle intermittent operation. Specifically, batteryless occupancy sensing devices must be careful to (1) optimize operation to make best use of available energy, (2) use ultra-low-power techniques and passive methods to perform the actual sensing and support the applications, and (3) be failure resistant, gracefully handling power failures and returning to deterministic states.

Energy harvesters as sensors: A sensing system traditionally consists of a dedicated sensor to gather data, along with some form of processing and communication, powered from a reliable energy source. While some existing solutions [24] similarly use light signals to power and gather information, they separate their harvester sensors from the photodiodes used for sensing with switches to separate the harvesting signal from the sensor signal. We propose an alternative to this approach by making use of the collected energy itself as data simultaneously. This approach gathers data by inferring the signal from variations in the harvested energy, instead of using that energy to power an explicit sensor. This approach is more challenging than the traditional or separating techniques since drawing from the harvested power supply over the lifetime of the system inherently impacts the signal that we are reading from for our data which can impact how the sensor data is interpreted and used.

For example, door-mounted occupancy sensors can harvest energy from indoor and ambient lighting using solar panels pointed towards the floor or other reflective surfaces. Concurrently, this energy is also a *signal* that can be processed to gain insight into the changing environment of the building, the movement of people and objects, or even the time of day. We can use this correspondence between energy and data to enable passive sensing and consequently, batteryless occupancy detection. If a door-mounted entry and exit sensor has solar panels that point down towards the floor, a person walking through the doorway will block some of the light, reducing the energy harvested at that point in time. This event can be tracked passively, effectively transforming the solar panels into zero-power sensors. This signal will be affected by the changing power draw of the system (an artifact of the I-V curves of solar panels) and will have a changeable resolution and magnitude depending on the incident light intensity. These factors make the overall signal noisy; however, careful signal processing in the energy constrained computational environment can provide useful information, freeing up energy that would otherwise have been consumed by an actively-powered sensor (like a PIR sensor or ultrasonic range finder).

Human and building confounds: Harvesting both energy and signal from solar panels introduces confounding factors from the variability of lighting in buildings, and the variability of people and their habits. Many buildings will have well-lit rooms bordering dim hallways, and vice-versa. Natural light may be abundant in some rooms, while others have only artificial light. Clothing, hair color, skin color, walking speed, and height will all affect and potentially change the readings on the solar panels. Humans also have a tendency to complicate data by not strictly walking in and out of a doorway—they like to linger, pass-by, and abruptly change direction under the sensor, for example. Ideally, it is a goal for all occupancy monitoring using energy harvesting to be robust enough to be able to handle these confounding factors, but all occupancy monitoring systems suffer from these confounding cases.

Batteryless occupancy sensing has never been done; but can take advantage of a key observation to provide reliable service—the reality that the applications' harvested energy can also be used a data stream that serves as a sensor. By taking advantage of the temporal locality of energy harvesting and data in occupancy sensing, we can build a long-lived sensor that detects and identifies the movement of people as they enter and exit rooms. In the following sections we discuss Ray, a novel sensing system that demonstrates the feasibility and utility of intermittently powered, energy-harvesting devices, for sensing in the sustainable future Internet-of-Things.

3 The Ray Design

Ray is a slim, batteryless, occupancy-monitoring sensor system mounted to the top of a doorframe. It is powered by energy harvested from two arrays of indoor solar panels pointed at the floor. The panels serve two roles: 1) energy harvester and 2) sensor. These panels gather **energy** for computation, sensing, and signaling while also providing the **signal** that Ray uses to detect when a person walks through the doorway in the form of variations in the harvested energy. Ray records the direction—entry or exit—of each doorway event and stores this information in non-volatile memory for later transmission.

Design Goals: Unpredictable power supplies and human behaviors make designing an intermittently-powered occupancy sensor challenging. We designed Ray to meet the following design goals.

1. **Availability:** Doorway events can occur at any time. While many intermittent sensors gather data opportunistically as energy is available, Ray is designed to conserve its harvested energy so that it is available to detect ephemeral doorway events, whenever they occur.
2. **Accurate direction:** In addition to detecting someone passing through the doorway, Ray uses angled solar panels to accurately determine their direction. This plays a crucial role in inferring the occupancy of rooms and buildings.
3. **Variable lighting conditions:** Indoor lighting conditions can change over time, due to human behavior and the relative movement of the sun. We have designed Ray to work in a range of different lighting conditions by using detection circuits that respond to changes in light level, independent of the absolute amount of light, as well as tuning mechanisms built into the prototype.
4. **Variable human characteristics:** An effective occupancy sensor should work well in spite of variations in clothing, hair, height, walking speed, and skin color. By focusing on changes in total reflected light, Ray is robust to these human variations.
5. **Form factor:** We want Ray to be easy to deploy, to fit unobtrusively inside a door frame, and avoid contact with doors (on frames with doors). We could harvest more energy by wrapping Ray around the doorframe, but the system would be more expensive, harder to deploy, and more likely to interfere with doors, while also changing the aesthetics of the doorway.

What Ray is not. We also want to be clear about what Ray is *not*. Ray is *not* a security device. Ray helps building owners and managers understand how people move through buildings, but it is *not* designed to thwart malicious behavior. We can easily trick Ray with a flashlight or reflective materials, and we can disable it completely by covering its solar panels or turning off the lights. Users looking to prevent shenanigans or tomfoolery should use a different device. Users looking for a long-lived, low-maintenance, best-effort batteryless occupancy sensor for monitoring normal behaviors should read on.

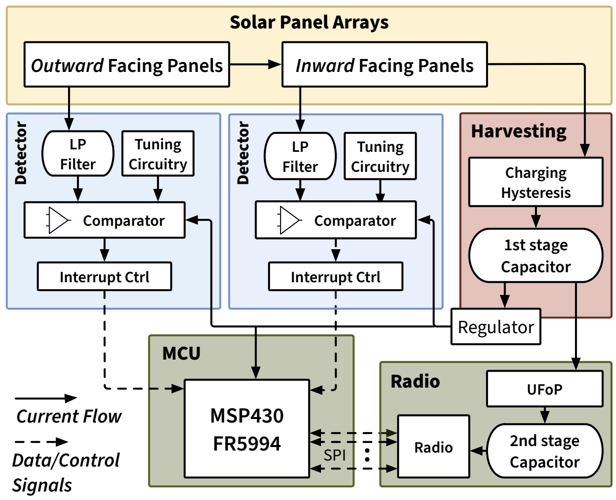


Figure 2: The Ray architecture overview. Ray uses the energy and signals from two sets of solar panels to both power the sensor and detect people passing into and out of a doorway. Two detector circuits each monitor the solar panels sets mounted in series in the doorway. One detector monitors a set of similarly facing panels, while the other observes the combined signal from all of the panels. On detection, the detectors wake up the MCU to process, log, or communicate occupancy information.

The Ray hardware architecture, shown in Figure 2, includes support for energy harvesting, event detection, computation, and communication. In this section, we describe these components and how they work together to meet Ray's design goals.

3.1 Energy Harvesting and Management

Ray takes advantage of the ubiquity of indoor light in homes and offices. Solar panels are mounted to the top of the door frame, pointing down toward the floor—half tilted 20° inward and half tilted 20° outward. Pointing the panels downward is not ideal for energy harvesting but effective for detecting doorway events and provides a slim, easy-to-deploy unobtrusive form factor. Tilted panels help Ray determine walking direction, since a person will affect one half of the panels before the other.

To maximize energy harvesting, we connect the two sets of solar panels—the inward-facing set and the outward-facing set—in series. A series configuration conveniently combines the two panel sets into a single power source that can be used without adding boost regulators or other power conditioning circuitry. This configuration makes it more difficult to analyze the two signals independently since they lack a common ground[2]. Instead, we measure the voltage of the outward-facing set alone, and the combination of the two sets. We could compute the inward panels' voltage by subtracting the two; however, we have found that we can skip this step and just compare the two measurements directly, as shown in Figure 3, to determine walking direction.

Ray uses federated energy storage [17] to power its microcontroller and peripherals. Harvested solar energy is fed into a common first-stage storage capacitor and then automatically federated to its one peripheral—a Texas Instruments CC1101 radio. Federating energy allows us to start detecting and processing events while saving up energy for more energy-expensive radio transmissions. It also improves harvesting efficiency and reduces the risk that the microcontroller will lose power due to a radio transmission.

3.2 Detection

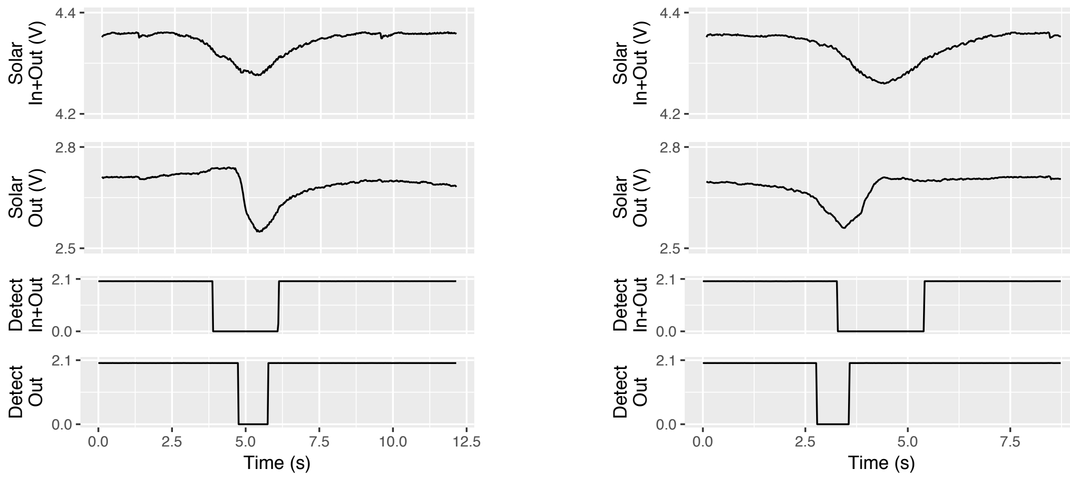
When someone walks under Ray, they block some of the reflected light hitting the solar panels. In Figure 3, the solar traces on top show how solar panel voltage changes during a doorway event.

In order to detect a doorway event, we could use an ADC to continuously measure the solar panel voltage over time and analyze those readings to detect the presence and, more importantly, direction of motion. Voltage levels and waveform shapes vary with lighting conditions, especially when one side of the doorway has more natural light. This approach would mean more complicated signal analysis and much higher energy consumption. Instead, Ray uses a **detection circuit** that wakes up the microcontroller when it detects a significant change in the solar panel voltage over a short period of time. This circuit consists of a passive first-order capacitive filter connected to a nano-power comparator—producing a square wave that transitions when the voltage increases or decreases faster than a set rate. These transitions trigger interrupts that help Ray detect when someone is passing through the doorway.

In order to determine movement direction, we use two detector circuits: one that detects change on the outward-facing panels and another that detects change on the combined inward- and outward-facing panels. When someone walks through the doorway, the detectors trigger at different times, depending on the walking direction, as shown in Figure 3. Ray compares the timing of these detector interrupts to distinguish incoming and outgoing events.

Removing light flicker. Many fluorescent indoor lights flicker at 60 Hz or higher—a much higher frequency than the events Ray detects. If not filtered out, fluctuations can confuse the detection circuit and produce false positive results. We add a low-pass filter to remove noise above 10 Hz from the solar panel signal.

[2]For a series connection, we connect the positive terminal of the first panel set to the negative terminal of the second.



(a) Walking in.

(b) Walking out.

Figure 3: These traces show example solar panel voltages and detector outputs over time when a person walks through a Ray-enabled doorway. The top traces show how the solar panel's voltages are deformed during the doorway event. The detector triggers are used to wake up the microcontroller and detect events and their direction. The angling of the panels cause the inward facing and outward facing detectors to trigger at different times depending on the direction the person is walking.

Isolating harvesting from sensing. If connected directly, Ray's harvesting and event detection circuits can potentially conflict in two important ways. First, the harvesting circuit stores harvested energy in a 100 µF capacitor—a size that ensures that Ray can store enough energy for short-term tasks and dampens the low-frequency voltage fluctuations that we need in order to detect doorway events. Second, short-term power spikes from interrupt service routines and other computation cause high-frequency dips in the solar voltage, which can confuse the detection circuits. We address both of these challenges by adding an additional low-pass filter between the detection and harvesting circuits. This isolates the solar panel from the load, and allows the solar panel voltage (after the initial flicker filter) to fluctuate over a wider range in response to doorway events with less interference from the storage capacitor, the microcontroller power draw, and the detector circuit power draw.

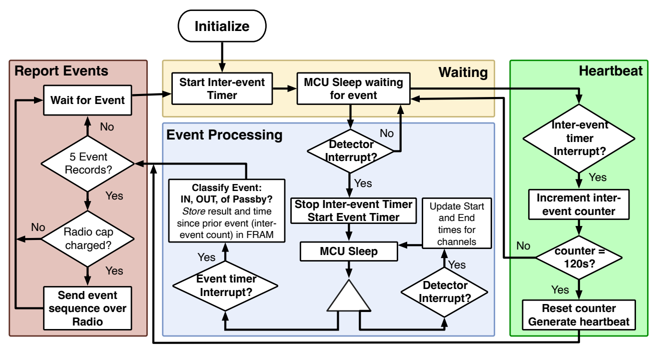


Figure 4: The Ray detection algorithm and system decision flowchart. The algorithm is composed of four parts that handle reporting doorway events, processing detected events, processing time between events and heartbeats, and idle waiting.

Detection algorithm: Ray's software works as shown in Figure 4. During normal operation, when no doorway events are detected, Ray's MCU remains in a sleep mode, only waking up to report heartbeats after two minutes of inactivity. While in sleep mode, the MCU only wakes up in two cases: 1) when its inter-event timer fires (this timer measures the time that has passed since the last reported event) and 2) when activity near the sensor triggers the detector circuits. A detector transition from a high state to a low state—due to a drop in harvested solar energy caused by a person walking through the doorway—will wake up the system to process an event. This initial wake-up marks the start of an event. The MCU records when the interrupt occurred, starts an event timer (6 seconds), and goes back into sleep mode. During the 6 s event, the MCU will wake up each time the detectors transition (from HIGH to LOW and vice versa) to record the length of time between each transition. A person may block light in many ways, and so multiple interrupts may fire during a single doorway event. When the event timer fires (after 6 s), the event is considered finished and event classification and recording begins.

Times are recorded for the first falling edge interrupt (start time) and the last recorded rising edge (end time) for both solar panel sets of inward and outward facing panels. When the event timer fires, the system has recorded both solar panel sets' start and end times, which are the features used to classify the event. For training, we collected features from 332 controlled events from two different people at three different

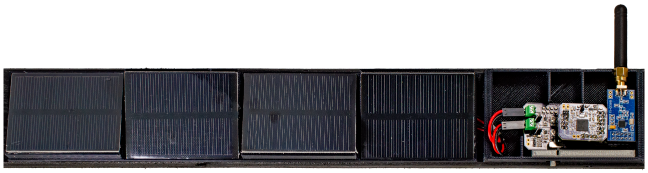


Figure 5: 3D printed solar panel and PCB housing enclosure with angled slots for solar energy harvesters and the Ray prototype PCB.

locations, a lab, an office, and a hallway. Using this data, we trained a 4-layer decision tree that classifies events as INs, OUTs, and PASS-BYs (when a person passes within a few feet of the sensor, but not under it.) When locations had a door, the door remained open for the collection. The trained decision tree is then used on Ray to classify observed events during deployments.

Between events, Ray uses an inter-event timer to estimate time between events. When the next event is detected, the inter-event timer is stopped and then the event timer is started for processing the event interrupts during the event window. A heartbeat record is generated when the time since the last event passes 120 seconds and is added to packet information to be sent with other records to the basestation. Once the event is processed and classified or heartbeat is generated, it is recorded in non-volatile memory along with the estimated time that has past since the last event or heartbeat record. The inter-event timer is then reset and the cycle continues.

3.3 Communication and Infrastructure

Ray's collected data are stored in non-volatile memory until the system has saved up enough energy to transmit. To reduce transmission cost, we summarize the last 5 recently-detected event records, sending the sequence of events in order of when they were recorded with the estimated time since the last event that was seen and the classification of that event as an in, out, passby, or heartbeat when 2 minutes of inactivity is recorded. We compute a CRC over each sequence summary and send the prior two sequence summaries in each packet to reduce the risk of losing information due to corrupted or missing packets and increase the likelihood that a basestation will receive the data. If the radio's capacitor isn't sufficiently charged, Ray will sleep and try again when the next event occurs. This pattern continues throughout the system's operational lifetime.

4 Implementation

In order to evaluate our approach, we implemented a prototype Ray sensor that includes custom hardware (shown in Figure 5), firmware for detecting and reporting doorway events, and a custom 3D printed doorway mounting system that holds the assembled PCB and solar panels in a slim profile (Figure 5).

Components	Cost per unit	Unit Cost for 1000
Solar Panels	\$ 1.95	\$ 1.76
Microcontroller (MSP430FR5994)	\$ 8.04	\$ 5.09
Wireless RF Transceiver (CC1101)	\$ 9.40	\$ 9.40
Components	\$ 17.02	\$ 6.63
*PCBs**	\$ 2.94	\$ 0.30
Entire Ray Prototype	\$ 39.35	\$ 23.18

Table 1: Breakdown of the Ray prototype costs at time of purchase for development.

Hardware: Our prototype hardware integrates four (4) RL-55x70 solar panels (70.00mm x 55.00mm) and custom printed circuit boards (PCB) housed in a 3D-printed plastic enclosure. The prototype uses an MSP430FR5994 microcontroller from Texas Instrument's (TI) FRAM line of ultra-low-power processors. The newest FRAM-based MSP430s have several advantages over previous models: lower sleep-mode currents, shorter wake-up latencies, and faster non-volatile FRAM. Entirely interrupt-driven and remaining asleep most of the time to conserve energy, Ray benefits from these improvements. The solar panels are connected in two angled series-connected banks, each consisting of two series-connected panels. We connect the panels in series to increase voltage to allow Ray to work in a wider range of lighting conditions and make doorway events easier to detect. Our panels—chosen to provide flexibility during prototyping—provide enough current to power the circuit with sufficient voltage levels for detection under a wide range of lighting conditions. Future designs will focus on miniaturization. The detector circuitry is made using nano-power comparators (TI TLV3691) and a passive RC filter network. In order to give us flexibility, the RC filter network is tunable using trim potentiometers pre-installation or digital potentiometers in deployment. The Ray PCB also has a TI CC1101 radio for communication. The hardware used in the Ray prototype, shown in Figure 5, is not prohibitively expensive or obtrusive.

The total cost of the current prototype at the time of purchase, including all PCBs, component parts, Radio modules, and solar panels is \$23.18 per unit if ordered in quantities of 1000[3]. The distribution of the prototype costs is shown in Table 1. The current prototype has several components that are meant to enable experimentation and testing (modular board design, jumpers, headers, test points, etc)—a commercial version of Ray will be dramatically cheaper and smaller.

[3]PCBs priced by seeedstudio.com.

Firmware: The Ray firmware implements the detection algorithm with a trained decision tree for event classification discussed in Section 3. Monitoring the interrupts from the detectors and deducing the direction of motion upon triggering are the main tasks of the system. The firmware is designed to be ultra-low power, even in active mode, and has low computational complexity, offloading the bulk of the detection to the hardware circuits. The Ray firmware is composed of 691 lines of commented C code, compiling to a 4459 byte image. This code size comprises only 1.7% of the available code space on the MSP430FR5994 (256KB), leaving ample room for implementing custom tasks, recognizers, or multiprogramming operating systems.

Mechanical Design: The 3D printed mounting system (shown in Figure 5) is made of PLA plastic and contains the PCB, solar cells, and necessary wiring connecting them. Ray's 3D printed enclosure measures 56.0 mm by 395.9 mm by 22.8 mm at its thickest point. The enclosure provides a nesting place for the solar cells, pointing downward. A simple slide-mounted cap was also designed to cover the PCB housing to help minimize distractions when deployed. The sensor could be minimized further by selecting smaller solar panels and by placing the PCB behind the panels rather than to the side of the panels. The angle of the solar cell slots is set such that half of the solar cells tend toward the entry, while the rest face toward the exit.

All software, firmware, hardware schematics and layouts, and 3D printed mounting system will be made freely available at publication time.

5 Evaluation

In order to evaluate the efficacy of our approach, we evaluated Ray in both controlled and in-the-wild deployments.[4] We also compared Ray to a similar commercial sensor during the in-the-wild deployments, which is discussed in greater detail in Section 5.2.4.

5.1 Controlled Experiments

We evaluated Ray's performance under controlled conditions in three phases to test different variables the system might encounter:

In **phase one** we tested Ray on multiple doorways, with different light levels, flooring, doorway heights, and doorway widths. For each test, we evaluated Ray's ability to detect someone passing through and determine the person's direction of movement. We also tested the system's robustness to variations in height, clothing, and hair color by including a diverse group of subjects. We tested on 7 different doorways with 9 people for a total of 881 different doorway events (each person walked through multiple times per doorway).

In **phase two** we tested the limits of the device, examining the factors that affect its accuracy, performance, and availability—including lighting conditions, walking speed, and short delays between doorway events. We also tested a variety of events that may be falsely detected as doorway events.

In **phase three** we explored the energy-harvesting ability and gather microbenchmarks of the energy consumption of the parts of the Ray system.

5.1.1 Methodology and Claims

The following experiments address the goals defined in Section 3. We address system availability (Goal 1) by demonstrating the low power draw of the system and the number of recorded doorway events (and the number of doorway events missed) for each doorway test. Further, we evaluated the accuracy in determining the direction (Goal 2) by observing how often Ray correctly determined walking direction. We explored variable lighting conditions (Goal 3) by testing the device under 7 different doorways and hallways with diverse lighting conditions, both typical and adverse. We address human variation (Goal 4) by evaluating different walking speeds and the effects of clothing and hair color/hair covering on detection patterns. We claim that form factor (Goal 5) is addressed by our prototype and slim mechanical design, described in Section 4.

We also tested the limits of the device, by varying different factors to see when the device stops working and exploring conditions that can confound the sensor. These tests cannot hope to cover all possible deployment conditions, but they do give a broad sense of the capabilities and limitations of Ray.

We gathered all electrical signal measurements, except where specified otherwise, using the Saleae Logic 16 logic analyzer[5] at a sampling rate of 5KS/s. The analyzer's high-impedance ADCs allow for unobtrusive signal monitoring. This sampling rate is sufficient to detect the types of slow-varying doorway events that human activities produce. We manually recorded the direction of each doorway event as ground truth to verify Ray's event detection accuracy, then compared the ground truth results with the results measured by the logic analyzer. We measured light intensity levels using a TSL2561 light sensor,[6] aligned to the same angle as the solar panels in both directions to get accurate light intensities falling on the panels.

Finally, we investigate the accuracy of Ray against our manually gathered ground truth (visually verifying a person entering or exiting the room) instead of comparing to another occupancy-detection system. We do compare Ray to

[4] Both controlled and in-the-wild deployments were approved by our Institutional Review Board.

[5] https://www.saleae.com

[6] https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf

Passageway #	Light Intensity (lux) Inside	Light Intensity (lux) Outside	Flooring Inside	Flooring Outside	Dimensions (cm) Height	Dimensions (cm) Width	Total Events #	Direction Accuracy(%)
Doorway 1	98*	93	Tile	Tile	202	88	119	94.1
Doorway 2	81*	74	Carpet	Tile	203	88	128	100.0
Doorway 3	60	57	Carpet	Tile	203	88	123	93.5
Hallway 4	71	69	Tile	Tile	236	243	123	97.6
Hallway 5	59	59	Tile	Tile	236	240	127	99.2
Hallway 6	56	62	Tile	Tile	236	244	118	89.8
Hallway 7	82	71	Tile	Tile	221	191	143	99.3

Table 2: Evaluation results with 9 test subjects having variable height, hair color, and clothing as described in Section 5.1.2. We tested 7 different doorways/hallways of varying light levels, dimensions, and flooring types, all of which had enough light to power Ray. We ran multiple people through each of these 7 passageways one at a time, noting the detection accuracy and how many of the detected events had correct direction. All controlled events were detected so we display the direction accuracy of those events above. All these results show that an adequately lit Ray occupancy sensor can accurately detect doorway events and their directions.

*Mixed Lighting — Combined natural and artificial light

a commercially available sensor in the later discussion on uncontrolled deployment.

5.1.2 Normal Operation

In order to evaluate how well our approach detects doorway events, we tested Ray across multiple different doorways with a diverse group of subjects. In these tests, we focused on detecting doorway events caused by a person walking *under* the doorway and accurately determining the direction of the person's movement.

Experiment Overview: We tested 9 different participants, with different physical characteristics—heights ranging from 5'4" to 6'4" and hair colors including blond, brown, black, and bald. Our test group included a wide range of clothing colors (light and dark) and a variety of head coverings (hats, beanies, and hijabs).

For this experiment, we attached Ray prototypes to the top of 7 different doorways and hallways. Table 2 describes the passageways, including light intensity levels, flooring type, and dimensions. For doorways with doors, the door remained open throughout the experiments. Due to differences in subject availability, we had seven of the participants walk into and out of the room or hallway at least 10 times in each direction on all 7 passageways. An additional two participants were asked to walk in and out of the doorways and Hallway 7 at least 5 times in each direction. When participants were able to complete more than the requested 10 passes, that additional data was collected as well. Some additional data was generated for these experiments since passersby would occasionally trigger the system. For the controlled data collection, we discarded events detected by the system when they were affected by someone other than the intended subject triggering the system, like a person passing by.

Results: The results of the controlled experiment, including 881 individual doorway events, are shown in Table 2. Each *event* consists of one person walking through one doorway one time. Participants walked through many different sides of the doorway, not just through the center each time, and they varied their entry and exit paths throughout the runs. Participants also choose their walking speed at each run; while most chose a natural walking pace, some did vary their speed occassionally. Ray successfully detected 100% of the 881 doorway events. Ray also determined the walking direction correctly for 849 (96.4%) of the events. Ray's performance was consistent across all test subjects, independent of human variations like height, gait, hair color, and clothing.

5.1.3 Factors affecting Ray's operation

In addition to testing "normal" walk-through conditions, in this section we examine factors that affect Ray's performance as an occupancy-monitoring sensor. It would be impossible to exhaustively study all possible variations, but we are able to explore how Ray reacts to a variety of conditions and behaviors that it will encounter in actual deployments. Specifically, we explored the following factors:

Light intensity: We tested Ray on a variety of doorways with varying lighting conditions, with results listed in Table 2. Since Ray's solar panels are sensitive to visible light and the IR spectrum, we used a TSL2561 sensor to measure both mixed signal (visible and IR) data along with purely IR data, and recorded the combined illumination value (in lux). Our current prototype is fully functional on doorways with light levels above 56 lux on both sides. An average room/hallway in an office-style setting has light levels around 70 lux, which is sufficient to power the Ray sensor. It is worth noting that we can customize Ray for exceptionally dark doorways either by increasing the number of solar panels without changing the working of the system itself, or by employing input booster circuits like the ones used in CleanCut [7].

Walking Speed: Ray detects people walking under doorways based on the changes they induce in the system's harvested energy supply. This means that if a person walks slowly enough, their movement should become imperceptible to the system. In order to evaluate this limit, we asked test subjects to walk under the sensor at different speeds. We used a metronome to which the subjects could match their steps in order to achieve a consistent, even speed. With extremely slow walking (slower than 1 ft/s), we did observe decreased accuracies. Ray occasionally detected a slow-moving doorway event as two events. No test subjects have yet been able to walk slowly enough to avoid detection entirely. We don't consider this to be a problem for Ray, since in practice, people don't often move at such slow speeds.

Door Width and Height: All doors (in Table 2) were around 203 cm tall by 88 cm wide. All hallways tested were 221 to 236 cm tall and 191 to 244 cm wide. In our experiments, the door width and height had no significant effect on the accuracy; however, the controlled experiments only tested when a single person went through a wide door at a time, and we did not control for participants walking through the middle or side of the door (they were asked to walk naturally) or their entry and exit angles around the passageway location.

Multiple people: Section 5.1.2 showed the ability of Ray to detect individual people walking through. A practical consideration would be to examine the performance of Ray when multiple people walk through.

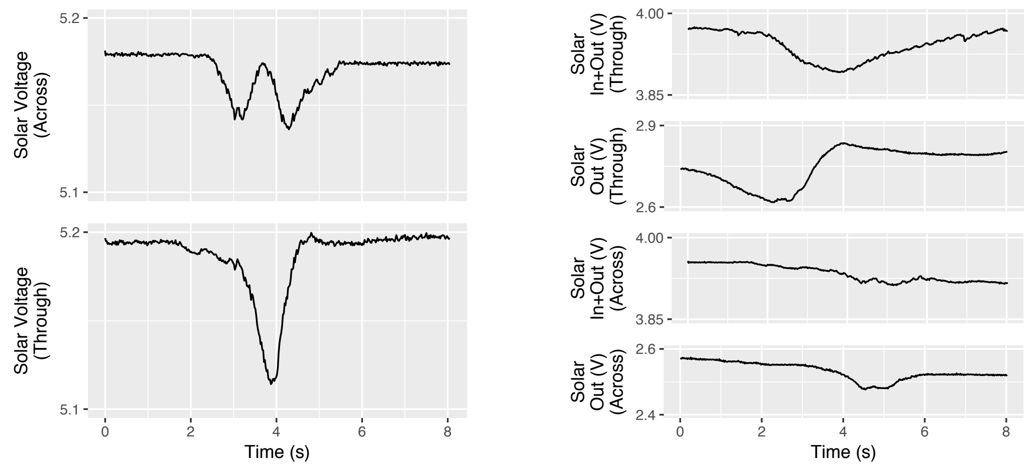
In order to evaluate this, we tested two subjects walking through doorway #1 with varying time delays between them. This gave us control over the time separation between two events, and allowed us to examine how closely can two people walk in without being detected as one, quite large person. We discovered that as long as two people have at least 3-4 seconds between them, Ray can accurately distinguish between them, but may incorrectly classify the direction. This limitation is introduced due to the time required by the solar panels to reset or stabilize before the next event can occur. A subsequent logical conclusion is that if two people walk side-by-side, *i.e.,* with zero separation between them, out current prototype is unable to detect them as two events.

The "Spotlight" effect: An interesting consequence of light-based detection is a problematic condition that can occur especially in mixed-light settings, when an intense low-angle light source dominates the illumination. This effect appears in the presence of a very focused source of light that dominates the illumination around the doorway, such as a spotlight or a west-facing window in late evening when the sun blazes directly through. When someone walks across the light source, even if they are far from the doorway, it can be detected falsely by Ray as someone walking through. Ray detects people based on a decrease in the harvested energy and momentarily blocking the spotlight can produce a significant decrease in voltage on both solar channels. Interestingly, we can see from Figure 6a that the raw output of the solar panels look sufficiently different for someone walking *across* the focused source as compared to when someone walks *through* the doorway in presence of a focused source. With further signal processing, Ray could distinguish these spotlight events so that such events would not cause false triggers.

Detection Range/Walking across, not through: Considering that Ray uses the blocking of light to detect a person, there will be an influence radius inside which a person starts affecting the sensor. If someone walks by either side of a doorway monitored by the Ray sensor and are within the radius, they will trigger the detector circuits and register as an event by Ray. We ran an experiment to determine this radius of influence where the subject was directed to walk by on either side of the doorway at increasing distances from the sensor. We started with a distance of 30 cm (1 foot) and went up to 152 cm (5 feet), in increments of 30 cm (1 foot). For each distance, we asked the subject to walk by multiple times and recorded how many false triggers were detected. An example of this is shown in Figure 6b. We have observed that under typically indoor lighting conditions for distances greater than 91 cm away from the doorway, there is a negligible chance of triggering false events.

It is interesting to note from Figure 6b that there is a distinguishable difference between this event as compared to someone walking through the doorway. Since they are walking only on one side of the doorway, their effect on both channels is not delayed by the angling of the solar panels, as is the case with walking through. As with the "Spotlight" effect, we should be able to extract this difference with further signal processing and learning.

Lingering in the doorway: Another situation that causes false triggers is when a person approaches the doorway, but simply pokes their head in. Upon evaluation, we discovered that as long as the person is poking their head in the doorway, the solar panel output remains at a lower level, and when they exit, it rises back again. Although the current system implementation isn't equipped to differentiate between someone passing through and someone lingering in doorway, there is a clear difference in the raw waveform outputted by the solar panel. This case is similar to Section 5.1.3 in terms of being distinguishable from a person walking through and with some careful, direct signal processing it is definitely possible to differentiate between the actual and the confounding case.



(a) These traces show the solar panel output in the presence of the "Spotlight" effect. The top figure shows the response when someone walks *across* the "Spotlight", while the bottom one shows the response when someone walks *through* the door.

(b) This figure compares a person walking *through* the doorway (top two traces) versus walking *across or by* the doorway on the outside. There is a clear delay between the two solar panel channels when someone walks through, whereas the change is reflected simultaneously when someone walks by.

Figure 6: Factors affecting Ray operation.

5.1.4 Microbenchmarks

The more effective Ray is at maintaining a low-power state when idling, the more available Ray is for detecting doorway events and monitoring occupancy. The energy requirements for detection and active computation must be kept low as well. Unlike intermittent computing systems, Ray must intentionally avoid power failures. We measured the current draw of our Ray prototype while it was mounted on doorway #1. The idle draw of the system was 7 µA, showing that Ray can survive in a doorway with minimal light and energy harvesting. We gathered other benchmarks of system energy performance in each of sysname's different operating modes. To seperate harvesting and consumption, these measurements were made after the MIC841 hysteresis chip. So, the actual power and energy is slightly higher (by 1.5 µA according to the datasheet).

Since Ray is event-driven, its actual power consumption varies depending on the activity underneath the sensor. As shown in Table 3 the idle power draw of Ray is low (7 µA). When timers or detector circuits trigger interrupts (maintenance events) the system draws 440 µW for a few µs. Computing walking direction, storing data, and transmitting results when an event ends is more expensive (1100 µW, on average). During typical operation, these higher-power events account for an insignificant fraction of the device's runtime, and the average power draw is often indistinguishable from the idle draw. Overall the energy consumption of the system is low, but could be further improved with careful tuning of resistance values, sleep states, and the analog circuitry.

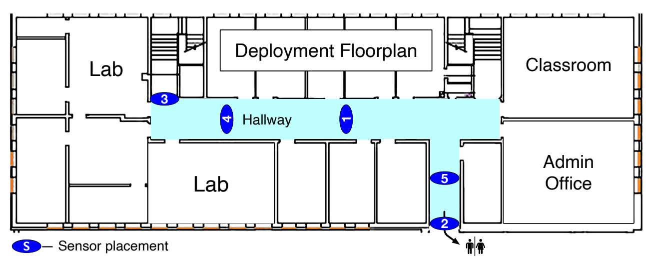


Figure 7: Ray deployment locations for the in-the-wild experiments. Each location features different lighting conditions as well as traffic patterns resulting from the adjoining labs, offices, and classrooms.

5.2 Rays in the Wild

In order to understand how Ray behaves in uncontrolled conditions, we deployed multiple Ray units along a hallway that connects offices, labs, classrooms, and bathrooms at locations shown in Figure 7 for a collective total of 64 days.

5.2.1 Experimental Setup.

We conducted in-the-wild experiments in two sessions. In the first, sensors were deployed at two locations (W1 & W2) for 24 days at the end of an academic semester and into the holiday class break. We observed events on 18 of the days

State	Avg. Current	Peak Current	MCU Active
Waiting (Sleep Mode)	7 µA	11 µA	✗
Maintenance Actions	200 µA	250 µA	✓
Doorway Event Handler	500 µA	700 µA	✓

Table 3: Microbenchmarks for Ray current consumption.

Location	Dimensions (cm)		Lighting	Traffic Profile	Days Deployed	Ground Truth Events
	Height	Width				
W1	243	235	Indoor	Light/Moderate	17	436
W2	221	180	Indoor	Light/Moderate	18	923
W3	202	149	Mixed	Moderate/High/Bursts	10	1067
W4	236	241	Indoor	Moderate/High/Bursts	10	1070
W5	239	190	Indoor	Moderate/High/Bursts	9	1294

Table 4: In-the-Wild deployment location descriptions and counts of ground truth events over the days deployed.

(only 17 for one of the sensors). In the second session, we deployed at three different locations (W3–W5) for an additional 11 days, with events recorded on only 9–10 days. This second session, at the start of a new semester, had heavier traffic, as shown in Table 4. The locations differed in light levels, width, and height, while providing a range of lighting and behavioral conditions. For example, the sensors near a classroom encounter multiple confounding cases like lingering and crowds passing under a doorway, while the ones near a lab or office are affected by lingering, spotlights, and occasionally crowds. We selected hallways in order to observe a wider range of natural traffic patterns, including multiple people walking together under the sensors. At each location, we installed a Ray sensor, a commercial ceiling-mounted EnOcean occupancy sensor [11], and a camera to provide a ground truth confirmation of hallway activities. All locations have tile flooring on both sides of the sensors and are typically well enough lit to transmit a packet once after at least 30 seconds have passed. This is usually ample time for the system to send a packet for every five detected doorway events or heartbeat (nothing has happened in two-minute) events, as events take at least 6 seconds each to process. We also deployed wall-powered basestations to collect the transmitted data from the batteryless Ray devices and EnOcean sensors. Each basestation is an Internet-connected Raspberry Pi with a CC1101 radio and an EnOcean receiver that receives packets and stores them in an SQL database for later retrieval. We deployed two base stations for redundancy—one would have been sufficient.

At each instrumented passageway, we also place a video camera that continuously collects ground truth information by recording the actual doorway events. We manually labeled all recorded events by watching the videos and annotating by storing the time and a description of each event—*in, out, pass-by*, as well as more complex cases like lingering, people changing direction under the sensor (u-turns) and multiple people passing in or out in a group. In order to make sense of the wide range of observed behaviors, we sorted the events into 3 different categories: simple events, multiple-people events, and complex events. Simple events include simple ins, outs, and very close pass-by events with only one person around the sensor within a 6-second window of time. Multiple people events involve multiple people that pass under the sensor going in the same direction within a 6-second window. These events range from 2 people walking side-by-side or close succession to 23 people all exiting at the same time when classes let out. All other events fall in the complex category, including lingering, changing directions underneath the sensor, and multiple people going in different directions under the sensor.

We compare these ground truth events against the sequence of events Ray detects. Ray send a message to the basestation once it records at least five events or heartbeats and has enough harvested energy. Ray generates heartbeat events after two minutes of inactivity. So, during long periods of inactivity, Ray will generate and send a packet every 10 minutes (consisting of 5 heartbeats) to let us know that it is still alive and has not seen any new events. This heartbeat frequency was selected for this experiment to help us distinguish between periods of inactivity (no detected events), dropped packets, and system failures. This frequency can also be adjusted to balance these liveness concerns with energy budget constraints.

Each network packet that Ray sends includes the sequence of the events and heartbeats, an estimate of time that has past between each event, their classifications, and a CRC computed over the recorded values to protect against packet corruption. As a simple redundancy mechanism in case of packet loss, each radio transmission includes the last two previously-sent packets along with the current packet.

For simplicity, Ray currently has no sense of absolute time—just relative inter-event time. In order to reconstruct the se-

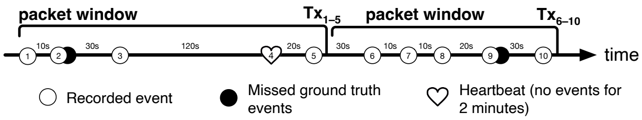


Figure 8: The ground truth event timeline of our in-the-wild deployment and how it is divided into packet windows based on the when the packet is recieved immediately after processing the fifth event record within the packet window. The heart at event 4 represents a heartbeat event generated by Ray when there have been no detected events for 2 minutes. The two examples of recorded event and missed ground truth events represent where two individuals walk through in quick succession, such that the sensor only identifies them as one event. This example provides a high-level view of how we compute the event statistics within a window.

quence of events, the time the basestation received the packet is used as a reference to match windows of events to the ground truth data. This time is closely correlated to the time of the packet's last event. So, we use these times to match ground truth events with the events recorded by the Ray sensors.

5.2.2 Data Collection and Analysis Method.

Each packet encodes the sequence of 5 events or heartbeats that Ray recorded and an estimate of time that has past since the last event or heartbeat recorded. A packet received at the basestation is matched with a series of ground truth events based on the packet's receive time, the estimated inter-event times, and the duration of Ray's event window (6 s). This helps account for minor human errors in labeling event times, as it is not always clear exactly when someone started affecting the sensor from video data. If two ground truth events match a packet, the one closest (in time) is chosen and mapped to the 5th event associated with this packet. Using the information stored in the packets about number of events and their estimated associated time between the events, we can estimate the match of the other ground truth events with their likely corresponding records in the packet record sequence to analyze hits or misses and if the classified direction was correct. Figure 8 shows a high-level example of the how ground truth events are divided by the packet window based on the received time of the packet in order to calculate correctly detected events and possible missed events, as well as when detected, if they were correctly classified. Each Ray packet has a monotonically increasing packet ID, which we use to detect packet losses. If a packet is dropped or corrupted, it can be reclaimed from the following transmission, which includes the last two packets sent.

5.2.3 Results.

Ray performed well when detecting activity that was taking place under the sensor and, when the activity was close enough, around the sensor. All events that happened under the sensor were detected, and throughout the deployment, we lost only one packet due to transmission errors. Based on the previous and subsequent packet numbers, we know the sensor recorded something but we can not verify whether the one ground truth event that occurred during this time was detected correctly. Table 5 shows the frequency of the different event categories that the sensors experienced while deployed and Ray's accuracy on classifying the simple in and out events where a single user walked under the sensor.

Simple Events significantly outnumbered the other two event categories across all sensor locations. Ray is specifically designed to detect simple one-person ins and outs, and these are the most common form of traffic we observed. Ray detected all of the simple events, and correctly classified their direction 83–95.5% of the time, depending on the sensor location. Nearly all misclassified simple events were misclassified as pass-by events, though a few out events were misclassified as in events.

As mentioned before, only one of the deployment locations (W4) was used for both gathering training data and our in-the-wild deployment. While this location (unsurprisingly) outperformed the others, the others were close behind—indicating both that the trained model was able to work well when used in different lighting conditions and that future Ray iterations might achieve small performance improvements by adapting the model in situ based on observed light conditions.

Ray also detected the **Multi-Person** and **Complex** events, including lingers, u-turns, and multiple people affecting the sensor in quick succession; however, the sensor was not always able to accurately estimate number of people passing by or their direction.

Multi-Person Events—multiple people pass together or in quick succession under the sensor in the same direction—typically result in undercounting. When the events completed within Ray's 6 s event window—common when two people were walking side-by-side—the events were reported as an in event or an out event, and the event directions were nearly always classified accurately (comparable to the accuracies reported for the simple events). When these events lasted longer than 6 s, Ray reported a group of multiple consecutive events, with the first event usually classifying the event direction correctly and subsequent events often misclassified when Ray's event windows often captured partial events.

Complex Events, including lingers, u-turns, near pass-bys, and multiple people passing the sensor simultaneously in opposite directions behaved similarly, producing a group of one or more consecutive events, except that the direction estimate for the first event in the group is also often incorrect. Another key difference is that some complex events can result in over-

Location	Ground Truth Events	Frequency of Events by Type			Simple Events		Accuracy		
		Simple	Multi-Person*	Complex	In	Out	In	Out	Total
W1	436	91%	4%	5%	201	190	71%	96%	83.12%
W2	923	89%	4%	7%	426	388	99%	77%	88.82%
W3	1067	83%	14%	3%	505	291	91%	88%	90.32%
W4	1070	83%	14%	3%	343	503	92%	98%	95.51%
W5	1294	85%	10%	5%	579	512	97%	93%	94.96%

Table 5: Ray in-the-wild deployment frequency of events by type at each deployment location and accuracy on how the system performed on classifying the simple ins and outs that were encountered over the deployment.

*Multi-Person Events — This category of events represents only multiple persons traveling under the sensor going in the same direction.

counting. For example, a single person lingering under a Ray sensor for a few minutes will produce multiple consecutive events. The longer the person spent underneath the sensor, the more of these events Ray would record.

Both Multi-Person and Complex events represent confounding cases for Ray—and challenging cases for technologies for monitoring human movement through buildings. While we plan to address them in future improvements (Section 7), for now their impact depends on traffic conditions. Under usual passageway conditions, a user wanting to count people can treat isolated events (events with more than a 6 s gap between them) as single person events with reliable direction estimates and end up with slight underestimates. Sequences of consecutive events (with no gap) can, for now, be treated as reliable activity measurements but not accurate people counts or direction estimates.

The first phase of our deployment (W1 & W2), during end-of-semester traffic conditions, saw fewer groups moving together and less overall traffic, and 89–91% of the observed events were simple in and out events with 9–11% confounding events (mostly two-person side-by-side events and some lingers). As traffic increased at the start of the following semester, locations W3–W5 saw an increase in overall traffic and confounding events increased to 15–17% of the total events. In spite of the traffic increase, both phases were dominated by simple events, and Ray provided information suitable for accurate people counting. Of course, in some extremely high traffic areas (e.g., the entrance to a sporting event or concert) we expect that Ray would have a high number of confounding events and behave like an activity sensor providing less information about individual people and their direction.

5.2.4 Commercial Sensor Comparison

As mentioned earlier, in order to compare Ray to its closest commercially available alternative, we deployed a batteryless commercial ceiling-mounted PIR occupancy sensor by EnOcean [11] alongside our Ray sensors on each passageway. While other similar sensors are available, we selected the EnOcean sensor because it was actually a battery-free commercial option that did not use rechargeable batteries and had transparent product specifications easily available online. This sensor also was more programmable for expermental repeatability and, at the time of purchase, more easily available in our country. This sensor is the powered by harvested energy, and uses ambient light (IR) changes to detect movement. Unlike Ray, this sensor only detects activity/occupancy (but no direction information).

EnOcean sensors send two types of packets: an *occupied packet* when an event is detected and an *unoccupied packet* after 10 minutes of inactivity has passed followed by every 30 minutes after that. If movement is detected, it sends an occupied packet to a receiver attached to the same basestation we use to receive packets from Ray. Like before, the basestation collects these packets and stores them in an SQL database for later processing and comparison with the ground truth data. Once EnOcean detects an event and sends an occupied packet, it will not detect any more events for the next 2 minutes. While this 2 minute blind-spot allows the device to recharge between radio transmissions, it is also a considerable disadvantage when compared to Ray's 6–7 second blind-spot. Figure 9 shows how the two sensors behaved in the face of a simple event and in the presence of slightly higher traffic during our deployment.

We use the data from both sensors to estimate the number of people that walk through the passageway, as shown in Table 6. With its smaller blindspot, Ray outperforms the EnOcean sensors in all cases, but especially during our second deployment (W3–W5) with increased traffic and more multi-person events. When profiling traffic through passageways, Ray not only provides higher resolution information, but it also provides additional direction information to building managers looking to accurately estimate traffic flows.

6 Related Work

Ray shares similarities with other occupancy-monitoring sensing systems, especially those that use doorway-mounted sen-

Location	Gnd Truth Total People	Ray Detected	Ray Accuracy	EnOcean Detected	EnOcean Accuracy
W1	452	434	96%	304	67%
W2	961	925	96%	613	64%
W3	422	308	73%	174	41%
W4	1440	1107	77%	551	38%
W5	1637	1316	80%	640	39%

Table 6: Comparison of performance between the EnOcean and Ray sensors at each location during the in-the-wild deployment. This comparison show how well each system was able to detect and monitor the number of people moving through a passageway. Due to high traffic and burst conditions that occur when class lets out, both systems are affected with being able to detect number of people passing through the passageway as EnOcean has a 2-minute blind spot after the first event is detected and Ray has a 6 second event window where it is processing a single event and misses multiple people traveling within that event window.

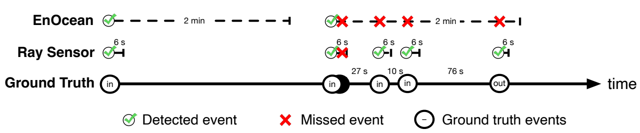


Figure 9: The ground truth event timeline of our in-the-wild deployment and how Ray and EnOcean sensors detect a sequence of events. This example provides a high level view of each systems detection accuracy and is based on actual data collected from our in-the-wild experiments.

sors. Ray also draws from literature on sensing systems that use harvested energy both as a power source for system components and as data signals. Recently, a batteryless network protocol [15] used occupancy-monitoring as a case-study to evaluate a new network protocol. With their solution, occupancy and direction were determined by transmitting the time between the two sensors placed adjacent to each other on the side of a doorframe when shadowed by an occupant in order to determine the direction a user traveled through the doorway. This case study observed a small group of users on a single doorway, focusing on network performance and latency rather than how well the system performs as an occupancy detector. This solution consumes more energy as it must transmit each time an event occurs in order to do the processing and classification off-device and may miss consecutive events due to slow charging times as occupants walk by the nodes. While this solution also using its power source as a sensor, it requires extra hardware (two nodes per doorframe) in order to perform occupancy monitoring applications. Ray, however, uses one piece of hardware with multiple panels mounted on top of a doorframe and processes event data on device, reducing overhead of sending timing information off device for processing, and reports multiple events to the basestation at once for energy savings, rather than needing to send each time an event occurs. The case-study did not provide enough information to directly compare the power-draw and performance of the two systems – making it difficult to compare head-to-head to Ray.

Occupancy-Monitoring Systems: Several different methods for the detection of occupancy and inter-room movement have been explored. Existing occupancy monitoring systems use ultrasound [20], imaging [35, 36], wearables [12], instrumented objects [4], structural vibrations [30], and opportunistic data leaked from existing meters and security systems [40]. These systems accurately detect occupancy (many provide other features like activity and person recognition); however, each suffers from the maintenance cost associated with battery-powered systems.

AURES [33] attempted to address this concern by using a rechargeable battery and an indoor solar panel. AURES estimates the number of occupants in a room by using wide-band ultrasonic signals. It needs to be installed in a central location on the room ceiling and near a light source to function properly. AURES, as an energy-neutral system, features an extended lifetime using energy harvesting to recharge a battery. However, all batteries wear out (usually in a few years) meaning replacement is inevitable. In comparison, Ray has the dual advantage of being both easy to install (on the doorway) and batteryless (lower maintenance).

Like AURES, EnOcean [11] and Leviton [23] are commercial ceiling-mounted occupancy sensors that are also powered by harvested ambient light and utilize passive infrared sensors (PIR) for detecting occupancy through motion detection. These sensors are equipped with wireless communication capabilities for transmitting the occupancy status (occupied/not occupied) of specific rooms or areas. This is useful in controlling the lighting, HVAC, and other electrical loads. In contrast, Ray uses the information present in harvested energy variations to detect individual doorway movements as well as the direction of those movements. This information can be used to improve utility decisions and help managers better understand how people use spaces and improve building layouts.

Another work proposes a battery-free camera powered by indoor ambient light to capture and transmit images via backscatter to a base station upon request [32]. Unlike Ray, this system uses a duty-cycle approach rather than an event-driven one for detecting an occupant. This results in either higher power consumption or many missed events.

CeilingSee [41] attempts to eliminate the extra power consumption of the monitoring tools by alternating existing LED lighting fixtures between light sources and sensors in a duty cycle manner. It uses reflected light and machine learning to distinguish between the fixed objects in the room and the room's occupants. CeilingSee offers a promising direction for new buildings, where custom lighting installations present an incremental cost. In contrast to Ray however, applying CeilingSee to legacy installations (old buildings) would be expensive, as this would include construction costs, computational infrastructure, and IT staff maintenance. CeilingSee could also put extra constraints on how a building can be lighted.

Recent work focuses on using multiple data sources that feed into a machine learning model to estimate the number of occupants in a building [9]. Using the number of connected WiFi devices to detect occupant count can provide coarse-grained information; however, it's severely limited by several possible cases, such as a single occupant connecting multiple devices, use of wired internet access, or not having any device connected to WiFi. This issue is addressed by monitoring utility data, such as water and electricity consumption, weather forecast, and building functions and size along with the number of WiFi devices. This combination works well at the building level. Unlike that, Ray is designed to monitor occupancy at room-level and communicate with other similar devices to deduce building-level occupancy. LOCI [28]uses data fusion from two types of sensors PIR and thermopile to localize occupants in the workspace and estimates their height. It is not batteryless and seems to be a power-hungry system since it consumes 460mW including packet transmission.

Doorway Occupancy Monitoring: The UVa Doorjamb sensor [20] enabled room-level tracking of people as they moved through a house, using ultrasonic range finders mounted above a doorway, pointed towards the ground. Doorjamb differentiates people by height and detects the direction of entry and exit into the doorway. A recent update—SonicDoor [22]—identifies occupants by sensing their body shape, movement, and walking pattern using ultrasonic sensors embedded in the sides and top of the doorway. SonicDoor also senses user behaviors like wearing a backpack or holding a phone. Doorjamb also used high-power sensors, wired power, and offline processing. Both systems depend on reliable power (wired power or batteries), and use high-powered sensors (ultrasonic range finders), in contrast to Ray, which uses energy harvesting and passive detection techniques to detect people walking through a doorway, providing room-level occupancy detection.

Energy as Data Sensing: Ray uses solar panels as both energy source and sensor simultaneously. This technique has been used in other systems for applications other than occupancy monitoring. Monjolo [10] measures the AC load consumption based on the harvested power from the AC load. Trinity [38] is designed to measure the airflow speed of air-conditioning based on the harvested power from piezoelectricity that is generated from the impact of airflow. DoubleDip [27] adapted this technique to monitor the water flow through a pipe using a thermoelectric generator as a harvester and sensor. Along with these, KEH-Gait [39] is designed for healthcare authentication and providing activity tracking. It does this by sensing the voltage level produced by two types of kinetic harvesters (piezoelectric and electromagnetic), which simultaneously also power the system. There has been another attempt to design a battery-free pedometer [21] by placing a piezoelectric harvester inside a shoe and estimating the number of steps based on the amount of harvested energy.

In addition, some indoor-sensing and ambient light-powered systems utilize solar panels as either a power source or sensor [3, 5, 24], but not both. SolarWalk [3] does use ambient light and a solar panel as its sensor, but doesn't harvest energy from the panel to power the system. It uses a PIR sensor to detect when a person is crossing the threshold and then records and transmits the raw solar panel data to be processed off-device for classification and identification of subjects. While it does point to an exciting direction of identifying users using solar data, it is limited in it's deployability as it is wall-powered, energy-expensive having to send off raw data each time there is an event, and processes the data off-device, which can expose privacy risks. SolarGest [26] and Ray share a similar method for using solar energy as an indicator for light ray interference, SolarGest is also batteryfree. However, SolarGest differs from Ray in application, implementation, and focus. SolarGest measures a small set of gestures (6) that are performed close to the solar panel; these controlled conditions are in contrast to Ray, which must deal with a wide range of confounding conditions and scenarios. Unlike SolarGest, Ray does all recognition in-situ, while SolarGest must rely on a backscatter communication channel with which it sends all data. This offline processing severely constrains the applicability of SolarGest. Li et al. [24] used photodiodes for both energy harvesting and sensing to recognize finger gestures on wrist- and head-worn wearables. In addition to significant application differences, they use a traditional duty-cycled ADC-based design that results in significantly-higher power consumption and requires more computation and orders of magnitude more energy storage (a 0.22 F supercapacitor compared to Ray's 100 µF capacitor). Using this approach for doorway event detection would require a significantly larger prototype. Also, unlike Ray, this work does not allow an energy channel to be used for harvesting and sensing at the same time for midair swipe-gestures—each photodiode unit is periodically disconnected from the harvestor circuit when needed for measurements. Ray's detector circuits enable simultaneous detection and harvesting in order to improve harvesting efficiency.

Batteryless, Transiently Powered Sensing: Recent work like InK [42], HarvOS [2], Mayfly [18], and Ratchet [37] have explored operating system and language-level support for developing applications easily on batteryless devices with frequent power failures. Others have focused on energy management and storage techniques, like Federated Energy [17], to improve system uptime and responsiveness. These systems inform our work, however, none has tackled the problem of batteryless occupancy monitoring.

7 Discussion & Future Work

In this paper, we demonstrated that we can monitor how people use buildings without running wires, without structural renovations, and without batteries. We have evaluated the performance of Ray as a batteryless occupancy sensor and identified corner cases that do sometimes confound the current version of the system. This section describes our future plans for making Ray more robust and reliable. We also present some ideas for extending this work.

Improving robustness and reliability: Ray in its current version depends on sudden changes in the solar panel outputs in a fairly binary manner. It triggers when there is change and doesn't when there isn't. This allows it to detect people walking through with high accuracy. However, it becomes susceptible to false positives as other events might also cause a sudden change in the solar panel, for example when someone walks by the side of the door. As discussed in Section 5, there is a visible difference between someone walking through a doorway and a false positive. One of our goals for future work is to explore the use of direct signal processing, allowing the microcontroller to analyze the entire waveform in software, rather than being limited to the hardware-provided detector interrupts. We expect that using an expanded range of signal features for classification will allow Ray to better differentiate between people walking through the doorway, false positives at the sensor's edges, and other complex events.

User perceptions of privacy: Occupancy monitoring is often privacy violating—cameras, audio, and other methods being examples. Privacy rights in the workspace have long been debated [29], with some workers reporting productivity suffered because of the perception of loss of privacy [34]. Even though Ray is privacy preserving, and incapable of gathering video, audio, or other personal information, we have not yet surveyed people who live and work with Ray in their room or office. We believe user perceptions of their privacy could inform both the design of future Ray prototypes and provide insight into this tension between privacy and real time occupancy monitoring. We plan to explore this in future work.

Adaptability: We plan to make the system more dynamic and flexible by providing adjustable thresholds to the detector circuit. This will equip Ray with the ability to tune its sensitivity to problematic cases, such as darker lighting conditions. Another way we aim to improve the performance and adaptibility of Ray would be to make use of learning algorithms. Our goal is to use learning for identifying different events and separating the true positives from false ones, subsequently improving accuracy and precision. We will introduce confidence indicators so that, even in cases where it is comparatively tougher to distinguish between those events, Ray will be able to attach a confidence level to its prediction, broadening the range of events it can identify. This is a feasible goal considering the evident difference between those events.

Network of Rays: Ray works as a standalone sensor, but we believe its true potential will be realized as a part of a network of similar sensors. Different Rays could exchange information to monitor occupancy on a larger scale and also to improve individual performance. For example, if one sensor detects a large amount of traffic heading into a hallway, but none of the other sensors detect activity, it is likely that there might be some other factor that is confounding the first sensor and this knowledge could be used to refine the learning model. Having a network of such batteryless sensors could also enable the deployment of a more sophisticated, energy-efficient communication model than simply broadcasting information opportunistically.

Additional sensors: We also plan to expand the system in terms of sensing abilities by adding more sensors. These sensors could provide various types of information such as RGB data, which could be used to semi-identify the person walking through. This would help assign some uniqueness to each individual so that we can better track their travel through rooms in a building without gathering identifiable information that would require additional security considerations to be added to the system. We could also opportunistically use an ultrasonic range finder in moments of high illumination to detect the height of the person passing through.

Ray can be expanded in many different ways, as demonstrated by these ideas.

8 Conclusions

This paper presents Ray, a batteryless, energy-harvesting doorway-mounted sensor system for room level occupancy monitoring. To our knowledge, Ray is the first batteryless occupancy-monitoring system in existence, and the first sensor device to simultaneously use its energy source—generated from opposing arrays of solar cells—both as a data signal for detecting doorway events and as an energy harvester that powers the system. Ray is built around a novel, tunable, detection circuit that watches the energy harvesting signal while the processor sleeps. We deployed Ray on 7 doorways and found that it can detect single persons moving through the doorways with a high overall detection accuracy of 100%. Our results show that Ray can differentiate between entry and exit of persons walking through the passageway for 96.4% of the detected events. We also evaluated different factors that affect the performance of Ray. While some events, like lingers, still confound the current version of Ray into generating false positives, we have demonstrated inherent differences between these events and true positives *i.e.,*someone walking through the door. This makes us confident that we can further improve the Ray system to make it robust to such events. We deployed several of these sensors in five different locations for an in-the-wild experiment over a total of 64 days collectively and found that the system detects events well when there are activities happening around the sensor to detect, even in the face of challenging confounding cases like lingering outside of a classroom door and heavy traffic flow when class lets out. We evaluated Ray microbenchmarks that demonstrate Ray is low power, and efficient, able to harvest enough energy to power all activities, intermittently, while providing quality of application. Ray represents a first step towards robust, reliable, and truly batteryless occupancy monitoring.

Acknowledgement

This research is supported by NSF award CNS-1453607. Any opinions, findings, conclusions, and recommendations expressed in these materials are those of the authors and do not necessarily reflect the views of the sponsors.

Artifacts

The artifacts associated with this paper are the PCB design and 3D prototype housing files used to create the sensor presented in this work as well as the software for training and implementing the deployed Ray functionality.

References

[1] Domenico Balsamo, Alex S Weddell, Geoff V Merrett, Bashir M Al-Hashimi, Davide Brunelli, and Luca Benini. Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems. *IEEE Embedded Systems Letters*, 7(1):15–18, 2015.

[2] Naveed Anwar Bhatti and Luca Mottola. Harvos: efficient code instrumentation for transiently-powered embedded sensing. In *IPSN*, pages 209–219, 2017.

[3] Md Fazlay Rabbi Masum Billah, Nurani Saoda, Victor Ariel Leal Sobral, Tushar Routh, Wenpeng Wang, and Bradford Campbell. Solarwalk: smart home occupant identification using unobtrusive indoor photovoltaic harvesters. In *Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 178–187, 2022.

[4] Michael Buettner, Richa Prasad, Matthai Philipose, and David Wetherall. Recognizing daily activities with rfid-based sensors. In *Proceedings of the 11th International Conference on Ubiquitous Computing*, UbiComp '09, pages 51–60, New York, NY, USA, 2009. ACM. URL: http://doi.acm.org/10.1145/1620545.1620553, https://doi.org/10.1145/1620545.1620553.

[5] Bradford Campbell and Prabal Dutta. An energy-harvesting sensor architecture and toolkit for building monitoring and event detection. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 100–109. ACM, 2014.

[6] Alexei Colin and Brandon Lucia. Chain: Tasks and Channels for Reliable Intermittent Programs. OOPSLA, 2016.

[7] Alexei Colin and Brandon Lucia. Termination checking and task decomposition for task-based intermittent programs. In *Proceedings of 27th International Conference on Compiler Construction*. ACM, 2018.

[8] Alexei Colin, Alanson P. Sample, and Brandon Lucia. Energy-interference-free system and toolchain support for energy-harvesting devices. In *Proceedings of the 2015 International Conference on Compilers, Architecture and Synthesis for Embedded Systems*, CASES '15, pages 35–36, Piscataway, NJ, USA, 2015. IEEE Press. URL: http://dl.acm.org/citation.cfm?id=2830689.2830695.

[9] Aveek K Das, Parth H Pathak, Josiah Jee, Chen-Nee Chuah, and Prasant Mohapatra. Non-intrusive multi-modal estimation of building occupancy. 2017.

[10] Samuel DeBruin, Bradford Campbell, and Prabal Dutta. Monjolo: An energy-harvesting energy meter architecture. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, SenSys '13, pages 18:1–18:14, New York, NY, USA, 2013. ACM. URL: http://doi.acm.org/10.1145/2517351.2517363, https://doi.org/10.1145/2517351.2517363.

[11] EnOcean. Ceiling mounted occupancy sensor, eosc (oem) [online]. 8 2018. URL: https://www.enocean.com/en/enocean_modules_902mhz/ceiling-mounted-occupancy-sensor-eosc-oem/.

[12] Kenneth P Fishkin, Matthai Philipose, and Adam Rea. Hands-on rfid: Wireless wearables for detecting use of objects. In *Proceedings of the Ninth IEEE International Symposium on Wearable Computers*, pages 38–43. IEEE Computer Society, 2005.

[13] Author Anonymized for Review. Master's thesis, 2018.

[14] Authors Anonymized for Review. Patent, 2022.

[15] Kai Geissdoerfer and Marco Zimmerling. Learning to communicate effectively between battery-free devices. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, pages 419–435, 2022.

[16] Josiah Hester, Timothy Scott, and Jacob Sorber. Ekho: Realistic and Repeatable Experimentation for Tiny Energy-Harvesting Sensors. In *Proc. 12th ACM Conf. Embedded Network Sensor Systems (SenSys'14)*, pages 1–15, Memphis, TN, USA, Nov. 2014. ACM.

[17] Josiah Hester, Lanny Sitanayah, and Jacob Sorber. Tragedy of the coulombs: Federating energy storage for tiny, intermittently-powered sensors. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 5–16, New York, NY, USA, 2015. ACM. URL: http://doi.acm.org/10.1145/2809695.2809707, https://doi.org/10.1145/2809695.2809707.

[18] Josiah Hester, Kevin Storer, and Jacob Sorber. Timely execution on intermittently powered batteryless sensors. In *Proceedings of the 15th ACM Conference on Embedded Networked Sensor Systems*, SenSys '17, New York, NY, USA, 2017. ACM.

[19] Josiah Hester, Nicole Tobias, Amir Rahmati, Lanny Sitanayah, Daniel Holcomb, Kevin Fu, Wayne P Burleson, and Jacob Sorber. Persistent clocks for batteryless sensing devices. *ACM Transactions on Embedded Computing Systems (TECS)*, 15(4), 2016.

[20] Timothy W Hnat, Erin Griffiths, Ray Dawson, and Kamin Whitehouse. Doorjamb: unobtrusive room-level tracking of people in homes using doorway sensors. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, pages 309–322. ACM, 2012.

[21] Haik Kalantarian and Majid Sarrafzadeh. Pedometers without batteries: An energy harvesting shoe. *IEEE Sensors Journal*, 16(23):8314–8321, 2016.

[22] Nacer Khalil, Driss Benhaddou, Omprakash Gnawali, and Jaspal Subhlok. SonicDoor: Scaling Person Identification with Ultrasonic Sensors by Novel Modeling of Shape, Behavior and Walking Patterns. In *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys 2017)*, November 2017.

[23] Leviton. Rf wireless self-powered occupancy sensor [online]. 8 2018. URL: https://store.leviton.com/products/rf-wireless-self-powered-occupancy-sensor-white-450-or-1500-sq-ft-coverage-available?variant=18216863235.

[24] Yichen Li, Tianxing Li, Ruchir A Patel, Xing-Dong Yang, and Xia Zhou. Self-powered gesture recognition with ambient light. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pages 595–608, 2018.

[25] Brandon Lucia and Benjamin Ransford. A simpler, safer programming and execution model for intermittent systems. In *Proceedings of the 36th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '15, pages 575–585, New York, NY, USA, 2015. ACM. URL: http://doi.acm.org/10.1145/2737924.2737978, https://doi.org/10.1145/2737924.2737978.

[26] Dong Ma, Guohao Lan, Mahbub Hassan, Mushfika B Upama, Ashraf Uddin, and Moustafa Youssef. Solargest: Ubiquitous and battery-free gesture recognition using solar cells. 2019.

[27] Paul Martin, Zainul Charbiwala, and Mani Srivastava. Doubledip: Leveraging thermoelectric harvesting for low power monitoring of sporadic water use. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, pages 225–238. ACM, 2012.

[28] S Narayana, V Rao, RV Prasad, AK Kanthila, K Managundi, L Mottola, and TV Prabhakar. Loci: Privacy-aware, device-free, low-power localization of multiple persons using ir sensors. 2020.

[29] Effy Oz, Richard Glass, and Robert Behling. Electronic workplace monitoring: what employees think. *Omega*, 27(2):167–177, 1999.

[30] Shijia Pan, Mostafa Mirshekari, Pei Zhang, and Hae Young Noh. Occupant traffic estimation through structural vibration sensing. In *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, pages 980306–980306. International Society for Optics and Photonics, 2016.

[31] Benjamin Ransford, Jacob Sorber, and Kevin Fu. Mementos: System Support for Long-Running Computation on RFID-Scale Devices. In *Proc. 16th Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS'11)*, pages 159–170, Newport Beach, CA, USA, Mar. 2011. ACM.

[32] Ali Saffari, Sin Yong Tan, Mohamad Katanbaf, Homagni Saha, Joshua R Smith, and Soumik Sarkar. Battery-free camera occupancy detection system. In *Proceedings of the 5th International Workshop on Embedded and Mobile Deep Learning*, pages 13–18, 2021.

[33] Oliver Shih, Patrick Lazik, and Anthony Rowe. Aures: A wide-band ultrasonic occupancy sensing platform.

In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, pages 157–166. ACM, 2016.

[34] Jeffrey M Stanton and Elizabeth M Weiss. Electronic monitoring in their own words: an exploratory study of employees' experiences with new types of surveillance. *Computers in Human Behavior*, 16(4):423–440, 2000.

[35] Thiago Teixeira and Andreas Savvides. Lightweight people counting and localizing in indoor spaces using camera sensor nodes. In *2007 First ACM/IEEE International Conference on Distributed Smart Cameras*, pages 36–43. IEEE, 2007.

[36] Ash Tyndall, Rachel Cardell-Oliver, and Adrian Keating. Occupancy estimation using a low-pixel count thermal imager. *IEEE Sensors Journal*, 16(10):3784–3791, 2016.

[37] Joel Van Der Woude and Matthew Hicks. Intermittent computation without hardware support or programmer intervention. In *OSDI*, pages 17–32, 2016.

[38] Tianyu Xiang, Zicheng Chi, Feng Li, Jun Luo, Lihua Tang, Liya Zhao, and Yaowen Yang. Powering indoor sensing with airflows: a trinity of energy harvesting, synchronous duty-cycling, and sensing. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 16. ACM, 2013.

[39] Weitao Xu, Guohao Lan, Qi Lin, Sara Khalifa, Neil Bergmann, Mahbub Hassan, and Wen Hu. Keh-gait: Towards a mobile healthcare user authentication system by kinetic energy harvesting. 2017.

[40] Longqi Yang, K. Ting, and M. B. Srivastava. Inferring occupancy from opportunistically available sensor data. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 60–68, March 2014. https://doi.org/10.1109/PerCom.2014.6813945.

[41] Yanbing Yang, Jie Hao, Jun Luo, and Sinno Jialin Pan. Ceilingsee: Device-free occupancy inference through lighting infrastructure based led sensing. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*, pages 247–256. IEEE, 2017.

[42] Kasım Sinan Yıldırım, Amjad Yousef Majid, Dimitris Patoukas, Koen Schaper, Przemyslaw Pawelczak, and Josiah Hester. Ink: Reactive kernel for tiny batteryless sensors. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, pages 41–53. ACM, 2018.