

Ray: Batteryless Occupancy Monitoring using Reflected Ambient Light

Anonymous Author(s)

ABSTRACT

Reliable and accurate 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, or are not privacy preserving. Furthermore, retrofitting existing infrastructures with wired sensors is prohibitively expensive.

In this paper, we present Ray, a *batteryless*, room-level occupancy monitoring sensor that harvests energy from indoor ambient light reflections, and uses changes in these reflections to detect when people enter and exit a room. Ray is mountable at the top of a doorway, allowing for detection of a person and the direction they are traveling at the entry and exit point of a room. We evaluated the Ray sensor in an office-style setting under mixed lighting conditions (natural and artificial) on both sides of the doorway with subjects exhibiting varying physical characteristics such as height, hair color, gait, and clothing. We conducted 651 controlled experiments in 6 doorways with 12 individuals and achieved a total detection accuracy of 97.38% and movement direction accuracy of 95.42%. We also describe an uncontrolled in-the-wild experiment using Ray sensors across 21 days collectively in three different locations. 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.

CCS CONCEPTS

- Computer systems organization → Embedded systems; Architectures;
- Human-centered computing → Ubiquitous and mobile computing systems and tools;

KEYWORDS

Occupancy, Batteryless, Intermittent, Energy harvesting

ACM Reference Format:

Anonymous Author(s). 2019. Ray: Batteryless Occupancy Monitoring using Reflected Ambient Light. In *Proceedings of The 17th ACM Conference on Embedded Network Sensor Systems (SenSys)*. ACM, New York, NY, USA, 13 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnn>

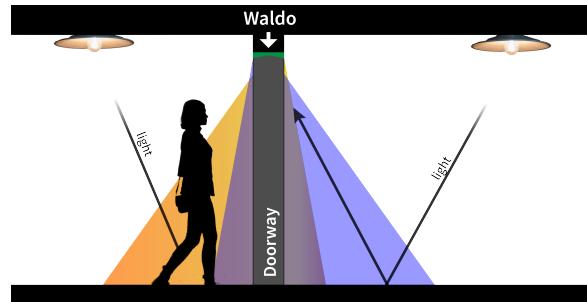


Figure 1: The overall system concept of Ray, a batteryless, energy-harvesting, doorway mounted occupancy tracking and person detection enabling system. This system uses reflective indoor lighting to both power the system and detect person entry and exit activity to a room.

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[16], images[28, 29], wearables[11], instrumented objects[3], structural vibrations[24], and opportunistic data leaked from existing meters and security systems[33]. 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).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SenSys, November 10–13, 2019, New York, New York, USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM... \$15.00

<https://doi.org/10.1145/nnnnnnnn.nnnnnnn>

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 decades¹ without wired power or batteries.

Like previous solutions [16], 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 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. 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 system, 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 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 [13]. 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, 25], consistent execution [5, 20], timekeeping [15], energy management [13], testing [12], and debugging [7] address key challenges and have enabled new and interesting applications. Examples of such applications include

tracking building and appliance energy consumption [4, 9] and monitoring greenhouses [13].

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 [16, 18]. 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. We propose an alternative to this approach by inferring the signal from variations in the harvested energy, instead of using that energy to power an explicit sensor.

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

¹Actual 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 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 some well-lit rooms bordering dim hallways, or vice-versa. Natural light may be abundant in some rooms, while others have only artificial light. Also, clothing, hair color, skin color, walking speed, and height will all affect and potentially change the readings on the solar panels. Any system that promises robust occupancy monitoring using energy harvesting must be able to handle these confounding factors.

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,

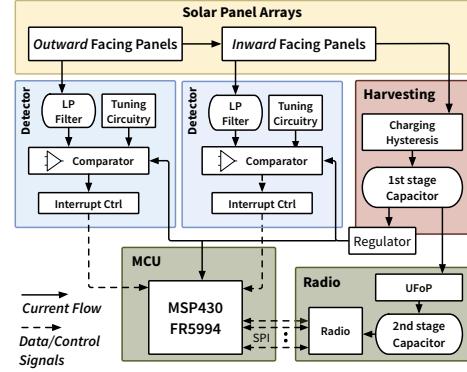


Figure 2: The Ray architecture overview. Ray uses the energy and signals from two sets of solar panels to both power the sensor and detect people passing into and out of a doorway. Two detector circuits each monitor the solar panels sets mounted in series in the doorway. One detector monitors a set of 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.

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.

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, but makes their signals more difficult to compare since they lack a common ground². 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 [13] 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 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.

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

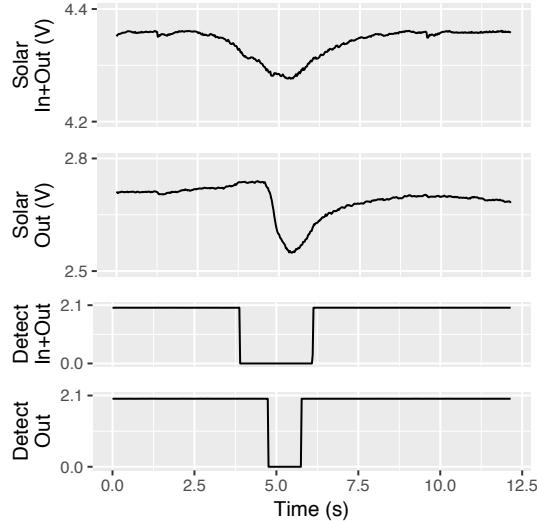
Isolating harvesting from sensing. If connected directly, Ray's harvesting and event detection circuits conflict in two important ways. First, the harvesting circuit stores harvested energy in a 100 μ F capacitor—a size that ensures that Ray can store enough energy for short-term tasks and dampens the 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.

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 deep sleep. While in deep sleep, the MCU is only triggered awake by the detector circuits going from high to low, due to a drop in harvested solar energy caused by a person walking through the doorway. This transition marks the start of an event. Once triggered, the MCU records when the interrupt occurred and starts a timer (a few seconds), then goes into sleep mode, waking up throughout the doorway event to capture the length of time between each detector's status change (from HIGH to LOW and vice versa). A person may block light in many ways, and so multiple interrupts may fire during a single doorway event. When the timer fires, the event is considered finished.

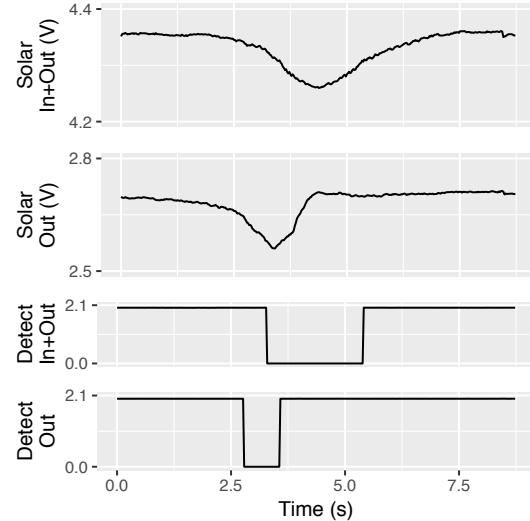
Times are recorded for the first falling edge interrupt and the last recorded rising edge for both solar panel groups. When the timer fires, the system has recorded both solar panel groups' start and end times, which are the features used to classify the event. For initial testing, we simply compare these times to determine the direction of the event (entry or exit), and the Ray stores the detected event in non-volatile memory. For deployment, we collected features from 296 controlled events from five different people at three different locations, and 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.) The trained decision tree is then used on Ray to classify observed events.

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 it. To reduce transmission cost, our current implementation summarizes recently-detected events, sending the number of ins, outs, and passbys seen in the last 5 or more events. We compute a CRC over each summary and send the prior two summaries with each summary to reduce the risk of losing information due to corrupt 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.



(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.

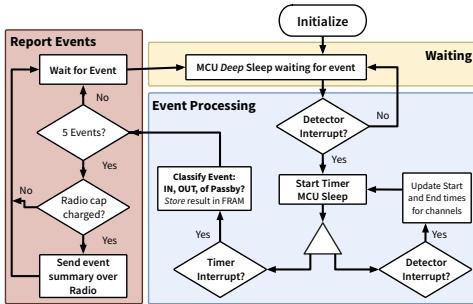


Figure 4: The Ray detection algorithm and system decision flowchart. The algorithm is composed of three parts that handle reporting doorway events, processing detected events, and idle waiting.



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

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).

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. 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.

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

Table 1: Breakdown of the Ray prototype costs.

The total cost of the current prototype at the time of submission, including all PCBs, component parts, Radio modules, and solar panels is \$23.18 per unit if ordered in quantities of 1000³. The distribution of the prototype costs are showed 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.

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 476 lines of commented C code, compiling to a 4285 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 plastics 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 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.

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 characteristics—light levels, flooring, doorway heights and doorway widths. For each test, we primarily evaluated Ray’s ability to detect someone passing through and determine the person’s direction

³PCBs priced by SeeedStudios.com.

of movement. We also tested the system’s robustness to human variation, like height, clothing, and hair color by including a diverse group of subjects. We tested on 6 different doorways with 12 people for a total of 651 different doorway events (each person walked through multiple times per doorway).⁴

In **phase two** we test 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 explore 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 itself and the number of times it caught 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 6 different doorways with diverse lighting conditions, both typical and adverse. We address human variation (Goal 4) by evaluating different walking speeds and the effect of clothing and hair color/hair covering on detection patterns. We claim that (Goal 5), concerning form factor, 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 generating 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 analyzer⁵ 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 the accuracy of Ray in event detection, then compared the ground truth results with the results measured by the logic analyzer. We measured light intensity levels using a TSL2561 Light-to-digital converter,⁶ aligned to the same angle as the solar panels in both directions to get accurate light intensities falling on the panels.

Finally, we note that it is difficult to fairly compare the performance of different occupancy-monitoring systems except in their accuracy. For example, CeilingSee [34] uses 16 devices to instrument a room, while Ray and SonicDoor [18] place one device in the doorway, and AURES [26] places a single device in the middle of a room. For this reason, 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.

⁴This study was approved by our Institutional Review Board.

⁵<https://www.saleae.com>

⁶<https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf>

Doorway #	Light Intensity (lux)		Flooring		Enough Light?	Total Events #	Detection Accuracy(%)	Direction Accuracy(%)
	Inside	Outside	Inside	Outside				
1	86	96*	Tile	Tile	Yes	162	93.82	98.02
2	86*	64	Carpet	Tile	Yes	63	90.48	78.94
3	71	55	Carpet	Tile	Yes	106	100	98.11
4	96	111*	Tile	Tile	Yes	106	100	98.11
5	55	55	Tile	Tile	Yes	112	99.1	94.59
6	55	71	Tile	Tile	Yes	102	100	96.08
7	40	71	Carpet	Tile	No	-	-	-
8	24	72	Carpet	Tile	No	-	-	-
9	24	55	Tile	Tile	No	-	-	-

Table 2: Evaluation results with 12 test subjects having variable heights, hair color, and clothing as described in Section 5.1.2. We tested 9 different doorways, from which 6 had enough light to power Ray. We ran multiple people through each of these 6 doorways, noting the detection accuracy and how many of the detected events had correct direction. These results show that an adequately-lit Ray occupancy monitor can accurately detect doorway events and their directions.

*Mixed Lighting – Combined natural and artificial light

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 twelve (12) different participants, with different physical characteristics—heights ranging from 5’4” to 6’4” and hair color ranging from blond to brown and black. Our test group included a wide range of clothing colors (light and dark) and a variety of head coverings.

For this experiment, we affixed the Ray prototype to the top of 6 different doorways. Table 2 describes the doorways, including details like light intensity levels and flooring type. For each doorway, we maintained test subject diversity, in order to characterize Ray’s performance, independent of the characteristics of individual subjects. For doorways with doors, the door remained open throughout the experiments. Each participant walked in and out of the room five times in each direction.

Results: The results of this experiment, including 651 individual doorway events, are shown in Table 2. Each *event* consists of one person walking through one doorway one time. Ray successfully detected doorway events and the direction people were traveling through the doorway, detecting a total of 634 events out of 651, giving an overall detection accuracy of 97.38%. Furthermore, it determined the walking direction correctly for a total of 605 events, which is 95.42% of the total events detected. Ray’s performance was consistent across all test subjects, independent of human variations like height, hair color, and clothing.

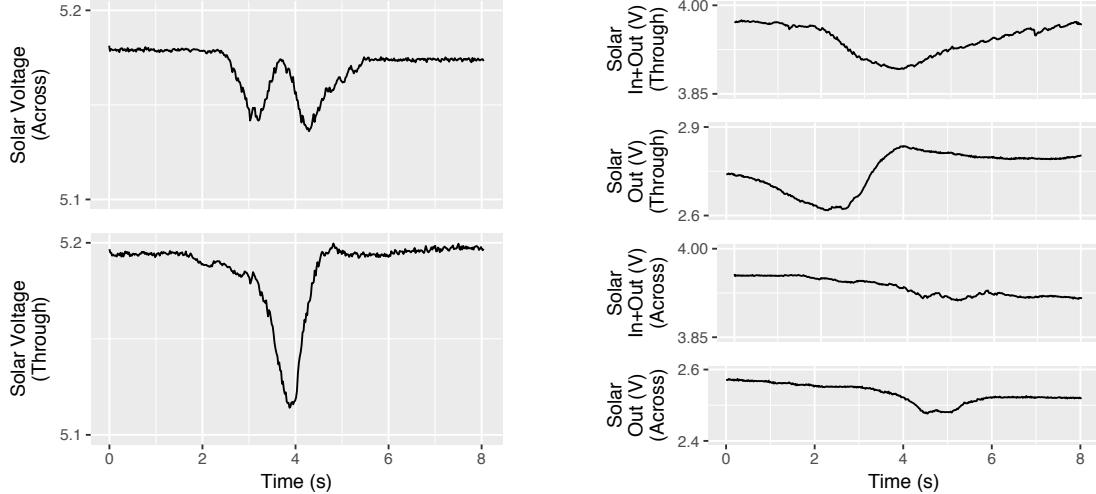
5.1.3 Factors affecting Ray’s operation. In addition to testing “normal” walk through conditions, in this section we examine factors that affect Ray’s performance as an occupancy-monitoring sensor. It would be impossible to exhaustively study all possible variations, but we are able to explore how Ray 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 outputs the combined illumination value in lux. As shown in Table 2, our current prototype is fully functional on doorways with light levels above at least 55 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 [6].

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 except door #4 (in Table 2) were 80 inches tall, door #4 was 87 inches tall. All doors except door #6 were 34.5 inches wide, while #6 was 58.5 inches wide. A typical interior doorway is 32 inches by 80 inches. In our experiments, the door width and height had no significant effect on the accuracy, however, we only tested when a single person went through a wide door, and we did not control for participants walking through the middle or side of the door (they were asked to walk naturally).

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



(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.

In order to evaluate this, we tested two subjects walking through doorway #1 with varying time delay 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. 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, the current setup of the system is unable to detect them as two events.

The “Spotlight” effect: A very interesting consequence of light-based detection is what we affectionately call the *spotlight effect*. 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 such a focused light source, even if it’s far from the doorway, it is detected falsely by Ray as someone walking through. This happens because Ray detects people based on a decrease in the harvested energy, which also occurs when someone momentarily blocks the focused source of light. 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. At present, Ray is equipped to detect events with good accuracy. With further signal processing, the difference between these events can be extracted so that such events will 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 harvested energy of the sensor. If someone walks by either side of a doorway monitored by the Ray sensor and are within the influence radius of the sensor, they will affect the sensor readings. 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 1 foot and went up to 5 feet, in increments of 1 foot. For each distance, we asked the subject to walk by multiple times and recorded how many false triggers were detected. Our findings are presented in Figure 6b. We have observed that for distances greater than 3 feet 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 and we attempted a stab at addressing this confounding condition further in the uncontrolled experiments that will be discussed later.

Lingering in the doorway: Another situation that causes false triggers is when a person comes up to 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

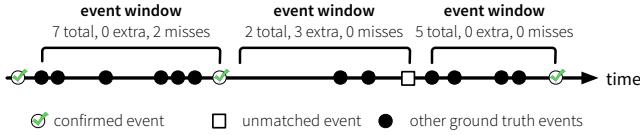


Figure 7: The ground truth event timeline of our in-the-wild deployment and how it is divided up into windows based on the anchor packets. This example provides a high level view of how we compute the event statistics within a window.

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.

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 $18\ \mu\text{A}$, showing that Ray can survive in a doorway with minimal light and energy harvesting. We gathered other benchmarks of system energy performance in each mode of operation Ray enters; the results of our experiments are shown in Table 3. These measurements were made after the MIC841 hysteresis chip, so the actual power and energy is slightly higher (by $1.5\ \mu\text{A}$ according to the datasheet).

Table 3 shows a low quiescent current where the MCU is asleep (the Idle state) of $7\ \mu\text{A}$ on average. Maintenance events (from timer firing or single detectors firing from noise or people passing by the doorway but not walking through) only require $140\ \text{nJ}$ to handle in active mode with the MCU running. The most expensive compute operation is when the detectors fire and the MCU must compute the direction and store the results to non-volatile memory: FRAM. This costs $4424\ \text{nJ}$. 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.

5.2 Rays in the Wild

We conducted an extensive in-the-wild deployment of multiple Ray units. The controlled experiments helped us understand some of the major limits of the system, but we also wanted to see how it fared in an in-the-wild deployment. For a real-life application evaluation, we installed Ray on two doorways and a hallway, gathering 21 days collectively of all-day occupancy data with visual (camera based) confirmation of ground truth.

5.2.1 In-Wild Experimental Setup. We chose locations—a lab door, a classroom door, and a hallway—that differ in light levels,

width, and height, while providing a range of lighting and behavioral conditions. For example, the classroom door encounters multiple confounding cases like lingering and crowds passing under the doorway, while the lab is affected by lingering and spotlights. Dimensions for these locations are reported in Table 4. All doorways in this deployment have tile flooring and are typically well enough lit to transmit a packet once for every five detected doorway events. Besides Ray devices we deployed two continuously powered basestations to collect the transmitted data from the batteryless Ray devices. Each basestation is an Internet-connected Raspberry Pi with a CC1101 radio that receives packets and logs them to a SQL database stored in the FLASH memory of the Pi. This data is easily retrieved for later classification of events.

At each instrumented doorway we also place a motion activated camera that collects ground truth information by recording the actual doorway events. We manually labeled all recorded events by watching the videos and annotating and storing the time and a description of each event—*in*, *out*, *pass-by*, as well as confounding cases like lingering and multiple people passing in or out.

We compare this source of ground truth against the events Ray detects. Ray attempts to send a message to the basestation only after gathering a total of five detected events (sometimes more, since there must be enough energy to send and events may be detected while waiting for energy to send). Ray summarizes the events, their classifications, computes and stores a CRC over the values to protect against packet corruption and sends the summary of the collection to a nearby basestation where it will be logged for analysis. Each packet also encodes a small legacy of the information received in the last two packets to guard against packet loss.

Ray does not currently report time information, so the time the basestation receives a packet is used as a reference to match windows of events to the ground truth data. Since Ray reports summaries after every five events, the basestation's receive time is closely correlated to the last event in the summary. We use these times to match likely event windows in the ground truth with received packets sent by the Ray sensors.

5.2.2 Data Collection and Analysis Method. Each packet (encodes a summary of at least 5 events) received at the basestation is matched to a likely ground truth event as long as the time between the two is less than 30 seconds. This helps account for human error in labeling event times as it is not always clear exactly when someone started affecting the sensor from video data. If two events match a packet, the one closest (in time) is chosen and mapped to the 5th event associated with this packet. Packets that map to a ground truth event are considered *confirmed packets*. All other packets that could not be mapped to an event (within 30s), are still valid packets that represent five events that Ray detected and must be considered. We insert these packets into the event sequence and mark them as *false positive* packets. The false positive packets may still contain valid data associated with the ground truth. The fifth event that generated the packet may have just been missed in the ground truth. These confirmed and false positive markers divide up the ground truth data into probable event windows in order to make a meaningful comparison with the packet summaries. We then gather statistics on the number of probable events associated with each packet. Confirmed packets are included in the associated

State	Avg. Power	Peak Power	Energy Cost	MCU Active
<i>Waiting (Deep sleep)</i>	7 µA	11 µA	-	✗
<i>Maintenance Actions</i>	220 µA	500 µA	140 nJ	✓
<i>Doorway Event Handler</i>	220 µA	500 µA	4424 nJ	✓

Table 3: Microbenchmarks for Ray energy consumption.

Doorway	Dimensions (in) Height	Dimensions (in) Width	Light	Total Days	Gnd Truth Events	Detected Events	Accounted Events	Extra Events	Missed Events	Recall
Lab	79.5"	34.5"	Mixed	8	445	735	354	388	91	74.02%
Hallway	76.0"	93.0"	Indoor	4	753	660	731	109	22	87.66%
Classroom	87.0"	35.0"	Mixed	9	213	300	167	107	46	68.84%

Table 4: In-the-Wild evaluation results for three different locations, from data collected over multiple days for each location. Ray effectively detected doorway events and other activities around the sensor, while troublesome cases like people lingering under/around the sensor can produce additional event information for a single event.

window counts, however, false positive packets, while still providing a meaningful anchor for the probable event window, did not have an event closely associated enough to add to the probable event window counts.

We reason about the number of events that Ray actually detects in these ground-truth based probable event windows by drawing on two characteristics of the system design. One, each packet encodes five events, and two, every packet will have a different packet id number, one greater than the last packet. Using this knowledge, we can figure out which packets, if any, we missed (what packet id numbers were absent on the basestation?) and how well Ray *detected events*, even if it may have classified events incorrectly. When all the ground-truth events in the event window were explained by the number of events in the anchor packet and missed packets, we considered the window to be accounted for. When they are not, we know that either events were recorded (on video) but not detected by Ray, denoted as *misses*, or Ray detected more events than were reported in the ground-truth data, which we called *extras*. Figure 7 shows a high level example of the how ground truth events are divided by the anchor packet time in order to calculate extras and misses. These values are aggregated over all probable event windows in the ground truth data.

The results, shown in Table 4, are mixed. Ray did a decent job of detecting when activity was taking place near the sensor, with few missed events. After inspecting the data, nearly all misses appear to come from three situations: (1) a person passed by the doorway, far from Ray but still visible to the camera; (2) multiple people passed by the door in close succession, seen as multiple events by the ground truth labelers, but only a single event by Ray; (3) packets were lost due to Ray power failures and sequences of dropped packets. Extra events were far higher than missed events in our data, primarily due to long and slow events—particularly common around the lab and classroom—many of which lasted minutes, were labeled as a single ground truth event but misclassified by Ray as many events (mostly passbys). Ray clearly has room for improvement when classifying confounding cases like lingering and other long events and high-traffic situations; however, even in these problematic cases, Ray serves as an effective activity sensor.

5.2.3 Results. After evaluating Ray event detection in general, we processed the data in a different way to reason about the system’s classification accuracy. Ray’s design makes it difficult to reason about true negative events. When there was no activity, no videos were recorded and Ray did not report events. This is intended behavior, but without discrete negative cases statistical measures are meaningless. Instead, we focus on the true positive rate, or Recall, as reported in Table 4.

We process the data slightly differently, in order to assess how the system classified events. We map all received packets to their closest matched ground truth event. As before, this marks the end of the collection of ground truth events that likely contributed to the creation of the associated packet. The matching ground truth window ends at the packet’s receive time and includes up to the prior 5 events as long as they are not included in another packet’s likely matched window. This means that some of our event windows contain fewer than five events, usually due to a linger event that is seen by Ray as multiple events. We compare these likely-matched event windows to number of ins, outs, and passbys reported in the received packets and compute the true positive rate (recall). Our classification accuracy ranged from 69-87% depending on the location, with highest accuracy in the hallway where people tended to walk more and linger less.

6 RELATED WORK

Ray is closely related to other occupancy-monitoring sensing systems—especially those using doorway-mounted sensor suites. Ray also draws from the literature on sensing systems that treat harvested energy both as energy and data signal; deriving application information from the energy source. SolarGest [21] is the most recent and closely related system. SolarGest 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.

Occupancy-Monitoring Systems: Several different methods for detection of occupancy and inter-room movement have been explored. Existing occupancy monitoring systems use ultrasound[16], imaging[28, 29], wearables[11], instrumented objects[3], structural vibrations[24], and opportunistic data leaked from existing meters and security systems[33]. 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 [26] attempted to address this concern by using a rechargeable battery and a 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 i.e. maintenance free.

Like AURES, EnOcean [10] and Leviton [19] 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 electric loads. On the other hand, Ray utilizes the information present in harvested energy variations to detect individual doorway movements as well as the direction of those movements. This information can not only be used in making smarter building utility decisions, it also provides a more detailed insight into area/room-wise usage of buildings in terms of occupancy count. This fine-grained occupancy information can possibly be used in optimizing the layout of a building and to provide higher flexibility in utility control.

CeilingSee [34] attempts to eliminate the extra power consumption of the monitoring tools by alternating existing LED lighting fixtures between being 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 rooms 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 [8]. 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 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 temperature, 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.

Doorway Occupancy Monitoring: The UVa Doorjamb sensor [16] 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 direction of entry and exit into the doorway. A recent update—SonicDoor [18]—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 of systems depend on reliable power (wired power or batteries), 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 [9] measures the AC loads consumption based on the harvested power from the AC load. Trinity [31] is designed to measure the airflow speed of air-conditioning based on the harvested power from piezoelectricity that generated from the impact of air flow. DoubleDip [22] 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 [32] 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 powers the system. There has been another attempt to design a battery-free pedometer [17] by placing a piezoelectric harvester inside a shoe and estimating the number of steps based on the amount of harvested energy.

Despite the fact that there is an indoor-sensing architecture that uses indoor solar-harvested power [4], this architecture does not implement the idea of using the energy harvesting source as the data sensor, as used in Ray. Ray is the first batteryless energy harvesting occupancy monitoring platform, that gathers both signal as well as energy from solar panels.

Batteryless, Transiently Powered Sensing: Recent work like InK [35], HarvOS [2], Mayfly [14], and Ratchet [30] 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 [13], 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. Our evaluation has presented the performance of Ray as a batteryless occupancy sensor and we also have identified corner cases that might confound the current version of the system. This section describes our future plans in terms of making Ray more robust and reliable. We also present some ideas for expansion of this project.

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's 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 add direct signal processing on the microcontroller so that it will be able to access the whole shape of the waveform, and will not be reliant on the binary nature of the detector interrupts. This will enable Ray to be more successful in identifying people walking through the doorway with minimal false positives.

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 [23], with some workers reporting productivity suffered because of the perception of loss of privacy [27]. 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 adaptability 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 is not meant to be a standalone system in that its true potential is realized as a part of a wider 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 of 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 6 doorways and found that it can detect single persons moving through the doorways with a high overall detection accuracy of 97.38%. Our results show that Ray can differentiate between entry and exit of persons walking through the doorway for 95.42% of the detected events. We also evaluated different factors that affect the performance of Ray. While some events 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 three of these sensors in three different locations for an in-the-wild experiment over a total of 21 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. 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.

REFERENCES

- [1] Domenico Balsamo, Alex S Weddell, Geoff V Merrett, Bashir M Al-Hashimi, Davide Brunelli, and Luca Benini. 2015. Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems. *IEEE Embedded Systems Letters* 7, 1 (2015), 15–18.
- [2] Naveed Anwar Bhatti and Luca Mottola. 2017. HarvOS: efficient code instrumentation for transiently-powered embedded sensing. In *IPSN*. 209–219.
- [3] Michael Buettner, Richa Prasad, Matthai Philipose, and David Wetherall. 2009. Recognizing Daily Activities with RFID-based Sensors. In *Proceedings of the 11th International Conference on Ubiquitous Computing (UbiComp '09)*. ACM, New York, NY, USA, 51–60. <https://doi.org/10.1145/1620545.1620553>
- [4] Bradford Campbell and Prabal Dutta. 2014. 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.* ACM, 100–109.
- [5] Alexei Colin and Brandon Lucia. 2016. Chain: Tasks and Channels for Reliable Intermittent Programs. OOPSLA.
- [6] Alexei Colin and Brandon Lucia. 2018. Termination Checking and Task Decomposition for Task-Based Intermittent Programs. In *Proceedings of 27th International Conference on Compiler Construction*. ACM.
- [7] Alexei Colin, Alanson P. Sample, and Brandon Lucia. 2015. 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)*. IEEE Press, Piscataway, NJ, USA, 35–36. <http://dl.acm.org/citation.cfm?id=2830689.2830695>
- [8] Aavek K Das, Parth H Pathak, Josiah Jee, Chen-Nee Chuah, and Prasant Mohapatra. 2017. Non-Intrusive Multi-Modal Estimation of Building Occupancy.
- [9] Samuel DeBruin, Bradford Campbell, and Prabal Dutta. 2013. Monjolo: An Energy-harvesting Energy Meter Architecture. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems (SenSys ’13)*. ACM, New York, NY, USA, Article 18, 14 pages. <https://doi.org/10.1145/2517351.2517363>
- [10] EnOcean. 2018. Ceiling Mounted Occupancy Sensor, EOSC (OEM). (8 2018). https://www.enocean.com/en/enocean_modules902mhz/ceiling-mounted-occupancy-sensor-eosc-oem/
- [11] Kenneth P Fishkin, Matthai Philipose, and Adam Rea. 2005. Hands-On RFID: Wireless Wearables for Detecting Use of Objects. In *Proceedings of the Ninth IEEE International Symposium on Wearable Computers*. IEEE Computer Society, 38–43.
- [12] Josiah Hester, Timothy Scott, and Jacob Sorber. 2014. Ekho: Realistic and Repeatable Experimentation for Tiny Energy-Harvesting Sensors. In *Proc. 12th ACM Conf. Embedded Network Sensor Systems (SenSys’14)*. ACM, Memphis, TN, USA, 1–15.
- [13] Josiah Hester, Lanny Sitanayah, and Jacob Sorber. 2015. 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)*. ACM, New York, NY, USA, 5–16. <https://doi.org/10.1145/2809695.2809707>
- [14] Josiah Hester, Kevin Storer, and Jacob Sorber. 2017. Timely Execution on Intermittently Powered Batteryless Sensors. In *Proceedings of the 15th ACM Conference on Embedded Networked Sensor Systems (SenSys ’17)*. ACM, New York, NY, USA.
- [15] Josiah Hester, Nicole Tobias, Amir Rahmati, Lanny Sitanayah, Daniel Holcomb, Kevin Fu, Wayne P Burleson, and Jacob Sorber. 2016. Persistent Clocks for Batteryless Sensing Devices. *ACM Transactions on Embedded Computing Systems (TECS)* 15, 4 (2016).
- [16] Timothy W Hnat, Erin Griffiths, Ray Dawson, and Kamin Whitehouse. 2012. Doorjamb: unobtrusive room-level tracking of people in homes using doorway sensors. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*. ACM, 309–322.
- [17] Haik Kalantarian and Majid Sarrafzadeh. 2016. Pedometers Without Batteries: An Energy Harvesting Shoe. *IEEE Sensors Journal* 16, 23 (2016), 8314–8321.
- [18] Nacer Khalil, Driss Benhaddou, Omprakash Gnawali, and Jaspal Subhlok. 2017. 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)*.
- [19] Leviton. 2018. Rf Wireless Self-powered Occupancy Sensor. (8 2018). <https://store.leviton.com/products/rf-wireless-self-powered-occupancy-sensor-white-450-or-1500-sq-ft-coverage-available?variant=18216863235>
- [20] Brandon Lucia and Benjamin Ransford. 2015. 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)*. ACM, New York, NY, USA, 575–585. <https://doi.org/10.1145/2737924.2737978>
- [21] Dong Ma, Guohao Lan, Mahbub Hassan, Mushfiqa B Upama, Ashraf Uddin, and Moustafa Youssef. 2019. SolarGest: Ubiquitous and Battery-free Gesture Recognition using Solar Cells. (2019).
- [22] Paul Martin, Zainul Charbiwala, and Mani Srivastava. 2012. DoubleDip: Leveraging thermoelectric harvesting for low power monitoring of sporadic water use. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*. ACM, 225–238.
- [23] Effy Oz, Richard Glass, and Robert Behling. 1999. Electronic workplace monitoring: what employees think. *Omega* 27, 2 (1999), 167–177.
- [24] Shijia Pan, Mostafa Mirshekari, Pei Zhang, and Hae Young Noh. 2016. Occupant traffic estimation through structural vibration sensing. In *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*. International Society for Optics and Photonics, 980306–980306.
- [25] Benjamin Ransford, Jacob Sorber, and Kevin Fu. 2011. 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)*. ACM, Newport Beach, CA, USA, 159–170.
- [26] Oliver Shih, Patrick Lazik, and Anthony Rowe. 2016. AURES: A Wide-Band Ultrasonic Occupancy Sensing Platform. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. ACM, 157–166.
- [27] Jeffrey M Stanton and Elizabeth M Weiss. 2000. Electronic monitoring in their own words: an exploratory study of employees’ experiences with new types of surveillance. *Computers in Human Behavior* 16, 4 (2000), 423–440.
- [28] Thiago Teixeira and Andreas Savvides. 2007. Lightweight people counting and localizing in indoor spaces using camera sensor nodes. In *2007 First ACM/IEEE International Conference on Distributed Smart Cameras*. IEEE, 36–43.
- [29] Ash Tyndall, Rachel Cardell-Oliver, and Adrian Keating. 2016. Occupancy Estimation Using a Low-Pixel Count Thermal Imager. *IEEE Sensors Journal* 16, 10 (2016), 3784–3791.
- [30] Joel Van Der Woude and Matthew Hicks. 2016. Intermittent Computation without Hardware Support or Programmer Intervention.. In *OSDI*. 17–32.
- [31] Tianyu Xiang, Zicheng Chi, Feng Li, Jun Luo, Lihua Tang, Liya Zhao, and Yaowen Yang. 2013. 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*. ACM, 16.
- [32] Weitao Xu, Guohao Lan, Qi Lin, Sara Khalifa, Neil Bergmann, Mahbub Hassan, and Wen Hu. 2017. Keh-gait: Towards a mobile healthcare user authentication system by kinetic energy harvesting.
- [33] Longqi Yang, K. Ting, and M. B. Srivastava. 2014. Inferring occupancy from opportunistically available sensor data. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*. 60–68. <https://doi.org/10.1109/PerCom.2014.6813945>
- [34] Yanbing Yang, Jie Hao, Jun Luo, and Sinno Jialin Pan. 2017. CeilingSee: Device-free occupancy inference through lighting infrastructure based LED sensing. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*. IEEE, 247–256.
- [35] Kasim Sinan Yıldırım, Amjad Yousef Majid, Dimitris Patoukas, Koen Schaper, Przemysław Pawełczak, and Josiah Hester. 2018. Ink: Reactive kernel for tiny batteryless sensors. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. ACM, 41–53.