PIRMedic: Physics-driven Fault Diagnosis for PIR Sensors

Ashish Kashinath
ashishk3@illinois.edu
University of Illinois at Urbana-Champaign

Sibin Mohan*
sibin.mohan@oregonstate.edu
Oregon State University

Akshay Nambi
akshay.nambi@microsoft.com
Microsoft Research, India

Sumukh Marathe†
sumukhm@umich.edu
University of Michigan

ABSTRACT

Passive Infra-Red (PIR) sensors are an integral part of modern living. They have diverse applications ranging from automatic lighting and heating control in smart buildings, towel dispensers in washrooms, security alarms (for intrusion detection) to even human detection robots (for search and rescue). Unfortunately, PIR sensors are prone to failures during deployment due to reasons such as environmental damage, incorrect installation and component degradation among others that can lead to incorrect or faulty data. Currently, such failures are typically detected using either : *(a)* heavily engineered data-driven, statistical approaches that can have high false positive rates due to unseen data patterns or *(b)* expensive methods that use additional hardware such as video cameras or a golden reference sensor. The second approach inhibits scalability.

In this work, we first create a *taxonomy* for the most common PIR sensor failures. We then present PIRMedic— a *physics-driven approach, implemented at the edge,* to detect the various classes of failures. We show that we can both *detect* and *diagnose* the failures in a PIR sensor using an intrinsic hardware signal *viz.,* the analog output from the pyroelectric element in the sensor. Using this hardware signal in conjunction with frequency analysis and supervised machine learning methods, we obtain a high accuracy of 98 − 99% in failure detection and diagnosis. We evaluate our methods using *real-world deployments* in three distinct locations, in different environment and usage conditions.

CCS CONCEPTS

• **Computer systems organization → Embedded and cyber-physical systems**; • **Hardware → Fault tolerance**.

KEYWORDS

Passive Infra-Red (PIR) Sensors, Fault Diagnosis, Fault Detection, Fault Isolation, Cyber-Physical Systems, Edge ML, Reliability

ACM Reference Format:
Ashish Kashinath, Sibin Mohan, Akshay Nambi, and Sumukh Marathe. 2021. PIRMedic: Physics-driven Fault Diagnosis for PIR Sensors. In *ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '21), November 17–18, 2021, Coimbra, Portugal.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3486611.3486658

1 INTRODUCTION

IoT applications deployed in smart homes, industrial control and healthcare systems require the correct functioning of a myriad set of sensors. The data from these sensors is used by IoT analytics pipelines to derive patterns, insights and make decisions. One of the most versatile and ubiquitous sensors used in numerous modern applications is a Passive Infra-Red (PIR) sensor. A PIR sensor is a discrete, electronic sensor that captures thermal radiation from human and human-like objects (*e.g.,* animals) to indicate occupancy in a region. They are used in a wide variety of environments, from building automation to safety-critical systems such as factories. Consequently, the correct operation of PIR sensors is critical to the quality and performance of many IoT deployments.

Given the nature of PIR deployments in the wild : for instance in remote, manually inaccessible, harsh conditions (*e.g.,* assembly line of a factory), they are prone to failures and degradation of physical components. These failures could result in false triggers or a failure to capture real movements *e.g.,* assembly line failures in a factory or inaccurate estimation of wildlife counts. Such sensor failures are often caused by intermittent sensor faults that can happen in any one of its internal physical components [4, 15].

Currently, failures in PIR sensor deployments are usually addressed in one of the following two ways : *(a)* using an additional, auxiliary sensor such as a CCTV camera combined with image processing algorithms (say to validate the occupancy data) or *(b)* using high-grade PIR sensors that possesses additional capabilities such as ultrasonic sensors and intricate optics. This makes the sensor deployment very expensive. In addition, deployments need to be refreshed periodically by replacing old sensors with new ones and the former are disposed off adding to both electronic waste and cost. A case in point is our university research building where the occupancy sensor deployed is a high-grade PIR sensor comprising both ultrasonic and infrared sensors, resulting in a *total recurring cost of close to $100000 every 10 years*[1].

Crucially, *current approaches do not diagnose the cause of failures or perform any analysis of sensor performance*, that can be valuable

[*] Work done while affiliated with University of Illinois at Urbana-Champaign
[†] Work done while affiliated with Microsoft Research India

[1] Our 4-floor building (with a total of 244 rooms and 24 aisle areas) has more than 300 occupancy sensors. Each PIR sensor costs $300 [10, 11] with it being refreshed every 10 years, a worst-case recurring cost of $90000 every 10 years, which is expensive.

BuildSys '21, November 17–18, 2021, Coimbra, Portugal

ACM ISBN 978-1-4503-9114-6/21/11…$15.00
https://doi.org/10.1145/3486611.3486658

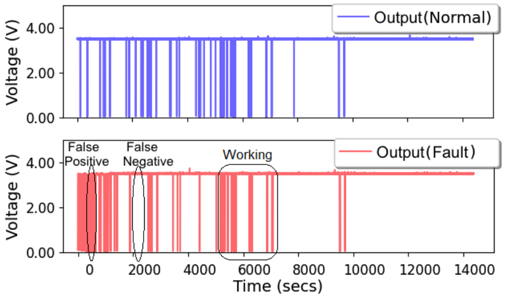


Figure 1: Sensor Data from a working PIR (top) and a faulty PIR (bottom). HIGH indicates no object, HIGH->LOW transition indicates object.

for IoT engineers during maintenance, repair and testing. For example, in one of our deployments, we observed that failed or degraded sensors can exhibit unpredictable behavior *e.g.,* they can work temporarily for a period of time and then fail sometimes. **Fig. 1** shows a plot of output data from the sensor vs. time for both a working sensor and a faulty sensor (defective lens) over a period of 4 hours. In this plot, a transition from HIGH⟶LOW denotes a person coming into the field of view and a value of HIGH indicates that there is no person. We observe that the faulty sensor works intermittently and at other times, can miss a person (false negative) or incorrectly flag the presence of a person (false positive). Missing a person can cause a potentially critical failure in applications such as door opening systems or emergency shutdowns on factory floors. We show that accurate detection of such failures is possible by *characterizing the internals of the sensor.*

Our solution, PIRMedic detects and diagnoses failures of PIR sensors by characterizing the physics of sensing viz., the fresnel lens optics and the pyroelectric effect. To keep the cost of the deployment low, PIRMedic is implemented on the edge platforms of the deployment and targets cheap PIR sensors. In this process, we *devised a taxonomy of key failures for aiding diagnosis and repair.*

Specifically, we demonstrate that a signal intrinsic to PIR sensors *viz.,* the *intermediate analog output from the pyroelectric element, referred to as A_{out}* (**Fig. 2**) can be used for fault detection and diagnosis.We utilize this signal to derive information about the reliability of the PIR sensor platform as well as perform an analysis into the different failure modalities. *To the best of our knowledge, this is the first edge-based, low-cost approach to use an intrinsic signal (A_{out}) for fault detection and diagnosis of PIR sensors.* We analyze the behavior of A_{out} in detail, under practical deployments, with realistic occupancy over a wide variety of observed failures.

Summary of Contributions:

(1) *From physics to failures:* We use the working of a PIR sensor to understand the process of object detection and infer the key points where failures lead to incorrect sensor data,

(2) *Failure taxonomy:* We systematize of key failures that can occur in a PIR sensor to develop a failure taxonomy.

(3) *Non-intrusive, online fault detection and diagnosis:* We present an online technique (does not require any disassembling), *implemented at the edge*, for failure detection as well as diagnosis by utilizing the intermediate, analog output from the pyroelectric element in the PIR sensor (A_{out}).

(4) *Insights from Real-world Deployments:* We show the efficacy of the proposed techniques using a deployment of 15 PIR sensors in practical occupancy scenarios (elevator, lobby of a building and at starbucks) over a period of 3 months.

2 RELATED WORK

Current solutions for failure detection and diagnosis in sensors fall into one of three categories – *(a)* Data-driven techniques, *(b)* Calibration-based techniques and *(c)* Fingerprint-based techniques.

Data-driven techniques. Prevalent research efforts have largely focused on data-centric approaches (rule-based or anomaly detection), where historical data of the sensor is analyzed and a fault is identified if the data is out of bounds of the expected behavior [25, 26]. Sharma *et al.* [28], proposed a multiplicative seasonal ARIMA time series model for fault detection, where the parameter captures periodic behavior in the sensor data. The downside of temporal analysis methods is that they are prone to false positives and are not feasible in long-term deployments. Wu *et al.* [33] used a spatial mining-based approach that uses spatial correlation between neighbouring sensors to detect anomalies. Additionally, techniques such as Ayadi *et al.* [2], Murphree [18] and Power *et al.* [25] require significant labeled data as it models only the environment and does not model the sensor physics. Estimation-based methods model normal sensor behavior leveraging spatiotemporal correlation and probabilistic models such as Bayesian or Gaussian distributions [12] and they work well in homogeneous environments.

We argue that for PIR sensors since the data is non-periodic and dependent on deployment scenarios (*e.g.,* people counting in starbucks, animal detection in forests, etc.), *it is non-trivial to detect faults by just analyzing sensor data.* Further, this requires significant manual efforts and tailor-made rules to detect faults, and can eventually have high false positives due to unseen data patterns.

Calibration-based techniques. These techniques rely on the presence of an additional (reference) sensor, either to perform periodic calibration [14, 32] or carrying additional information in a different domain space. *E.g.,* using a camera to cross-check the data of a PIR sensor [30]. Numerous algorithms have been developed for performing calibration such as blind calibration [9, 31], collaborative calibration [27, 34] and transfer calibration [35]. However, using reference sensors is an expensive strategy that lacks scalability for large IoT deployments or those in developing countries.

Fingerprint-based techniques. Chakraborty *et al.* [4] develop a sensor signature *viz.,* "Fall Curve" that measures a sensor's voltage response when the power is turned off, to detect faults in periodic on-off based analog sensors. Similarly, Tambe *et al.* [29] show a variant of Fall curve, i.e., "Fall time" of the analog signal can detect faults and drifts in phototransistor components of the sensor. However, both Fall Curve and Fall time signatures does not work on certain types of sensors including PIR, that operate under low voltages and are event-triggered. Marathe *et al.* [15] show that analyzing the current profiles of a digital sensor can provide insights into component failures especially in electro-mechanical sensors. However, this approach is limited to power-hungry digital sensors.

PIR sensors are challenging since in addition to being discrete-valued, the internal analog signal output from the pyroelectric sensor has a very low value, often oscillating between 1–1.8V. We show that the intermediate analog output (A_{out}) from the PIR sensor carries interesting insights on the characteristics of the sensor.

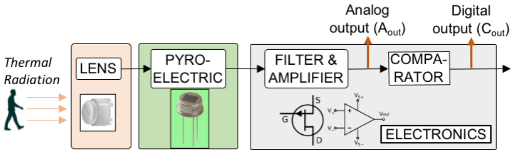


Figure 2: Internal components of a PIR sensor.

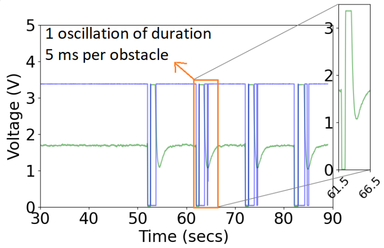


Figure 3: Output signals A_{out} (Green) and C_{out} (Blue) during sensor operation. A_{out}, (magnified) shows oscillations when an obstacle is detected.

Narayana *et al.* [19] leveraged this signal by developing a customized sensor array for performing localization and studying object metrics. However, Narayana *et al.* [19] do not explore reliability.

3 PIR SENSORS

In this section, we briefly describe the working of a PIR sensor and then present the potential failure scenarios in PIR sensors.

3.1 Background

A PIR sensor internally comprises of 3 subsystems as shown in **Fig.**2 – *(a)* a lens, *(b)* a pyroelectric, and *(c)* an electronic subsystem. These subsystems act in sequence *i.e.,* Lens → Pyroelectric → Electronic to perform the end-to-end sensing process as we describe next.

Lens Subsystem A plastic fresnel lens [1] is used to capture and focus thermal radiation into an optical filter that is aimed at a pyroelectric element. Fresnel lenses have a large capture area (aperture) and are used to concentrate radiation into a narrow beam.

Pyroelectric Subsystem comprises of an optical filter and a pair of pyroelectric elements. The optical filter is designed to filter out thermal radiations from wavelengths outside the human range (i.e., 5μm to 10 μm), which is then incident on the pyroelectric elements. The pyroelectric elements convert the thermal radiation into an electrical voltage signal, a process known as the *pyroelectric effect.* The output of the pyroelectric subsystem is an analog signal that is sent to the electronic subsystem.

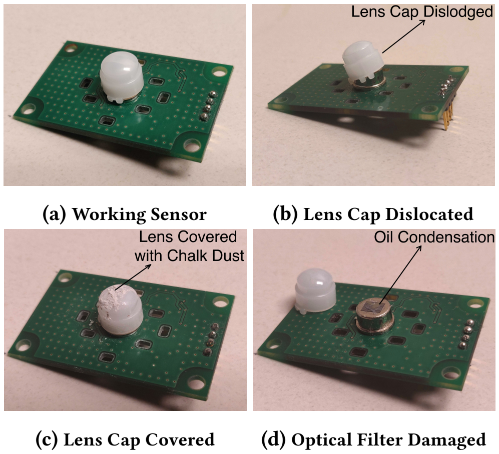
Electronic Subsystem consists of a filter (RC filters), amplifier (JFET or OpAmp) and comparator (OpAmp) circuits. Outputs from pyroelectric subsystem arfiltered to remove noise, amplified to increase its magnitude and finally sent to a comparator to convert analog signals to discrete signals. This *discrete signal is HIGH when there is no motion and goes HIGH→LOW when motion is detected.*

3.2 Analysis of Output Signals of a PIR Sensor:

There are 2 output signals from a PIR sensor as shown in **Fig.** 2 :

(1) **Final discrete output** of the electronic subsystem (C_{out}).
(2) **Intermediate analog output** just prior to the discretization process (A_{out}).

We interface a PIR sensor to an Arduino microcontroller to analyze C_{out} and A_{out} during operation. **Fig.** 3 is a plot of A_{out} and C_{out} for a period of 60 seconds, both in the presence and absence of an obstacle. The y-axis represents the voltages seen at the output and x-axis represents time. As expected, C_{out}, marked in blue goes LOW when an obstacle comes into the field of view and stays HIGH otherwise. On the other hand, A_{out} measures around 1.8 V when there is no obstacle and produces an oscillation when an obstacle comes into the field of view (shown magnified). We next study the various failures in a PIR sensor and its impact on sensing physics.



(a) Working Sensor **(b)** Lens Cap Dislocated

(c) Lens Cap Covered **(d)** Optical Filter Damaged

Figure 4: Some sensors used in our study. Note that the failures are not always visually perceivable as the sensors are typically inaccessible.

3.3 Failures in PIR Sensors – A Taxonomy

Broadly, failures could occur either on the lens, the pyroelectric element or the electronics. We *analyze and categorize practical, common and most frequent* failures as gathered in discussions with IoT engineers and technicians who install and maintain PIR sensor deployments. We developed a new taxonomy for such failures, as summarized in **Table** 1.

3.3.1 ***Failures in the lens subsystem.*** The lens is geometrically constructed to precisely focus the thermal rays onto the pyroelectric element. As a result, failures affecting the optical integrity of the lens can result in loss of precision for focusing the thermal rays. We observe three types of failures here, termed Class I–III failures. Lens dislocation (Class I): As the lens is stuck on the sensor board with an adhesive or machined to fit over the pyroelectric element, it could get dislodged from its place or completely fall off the sensor board due to factors such as thermal expansion or physical impact.

Table 1: Taxonomy of Failures in a PIR Sensor

FAILURE	DEFINITION & IMPACT
Lens Subsystem	
Lens dislodged (Class I)	Lens cap suffers partial or complete dislocation *e.g.,* physical impact with a foreign object, degradation of bonding, *etc.*
Lens deformed (Class II)	Lens cap suffers physical damage in-place *e.g.,* deformation, puncture, *etc.*
Lens covered (Class III)	Lens cap gets physically obstructed by foreign particles *e.g.,* dust, paper or tape
Pyroelectric Subsystem	
Optical filter damage (Class IV)	Damaged by environmental factors *e.g.,* oil condensation
Electronic Subsystem	
Electronic faults (Class V)	Hardware failures *e.g.,* short circuits, floating outputs *etc.*

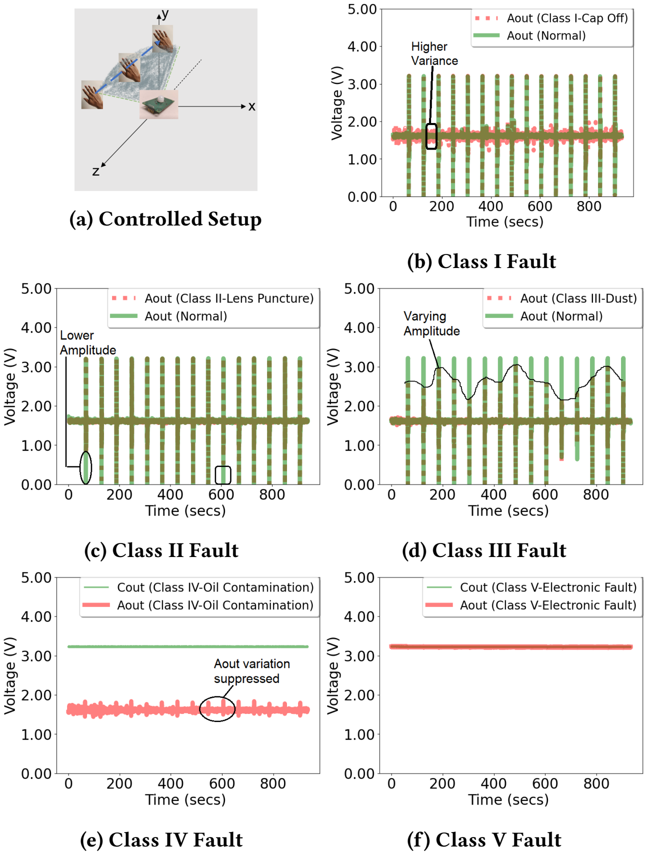
Lens deformation (Class II): As the lens is made of flexible, 0.4mm thick high density plastic [20], it is susceptible to deformation by physical damage. This can alter the curvature or puncture the lens. Lens hindrance (Class III): Dust particles (*e.g.,* factory floors, building construction) or common objects such as paper and plastic tape can absorb the thermal radiation resulting in reduced to no radiation falling on the pyroelectric element. The deposition of particulate matter also causes dispersion.

3.3.2 *Failures in the pyroelectric subsystem: Optical Filter Damage.*

The optical filter and pyroelectric elements are prone to degradation with exposure to high temperature and humidity. As the optical filter is carefully calibrated to trigger in the region of human motion, it is susceptible to damage due to environmental factors such as heat sources (*e.g.,* room heaters, air-conditioners) and humidity (*e.g.,* oil condensation) as it affects the perceived temperature of the object. We name these as Class IV failures.

3.3.3 *Failures in the Electronics.*

On-board electronics consisting of filter, amplifier and comparator circuitry are prone to failures such as shorted or floating pins that we refer to as Class V failures. **Summary:** Failures in PIR sensors can occur in either the lens subsystem, the pyroelectric subsystem or the on-board electronics, each of which manifest differently in the underlying physics.



(a) Controlled Setup

(b) Class I Fault

(c) Class II Fault

(d) Class III Fault

(e) Class IV Fault

(f) Class V Fault

Figure 5: (5b - 5f) A_{out} waveforms for sensor faults of Classes I–V. Note that in Class I – III faults, despite the obstacle still being detected, A_{out} provides indication of underlying abnormalities in the sensor.

4 CHARACTERIZING FAILURES USING AN INTRINSIC SIGNAL

Our hypothesis is that the intermediate analog signal (A_{out}) from the pyroelectric element captures information that is critical to detect various failures. We now describe how this intrinsic signal and its underlying physics is useful to characterize failures.

4.1 A_{out} signal to detect failures in PIR sensors

As described in Section 3.2, there are two output signals from a PIR, *viz.,* A_{out} and C_{out}. The latter is a discrete signal typically used for detecting human presence and is derived from A_{out}. To show the utility of A_{out}, we conducted numerous controlled experiments where we manually injected commonly seen faults (**Fig.** 4).

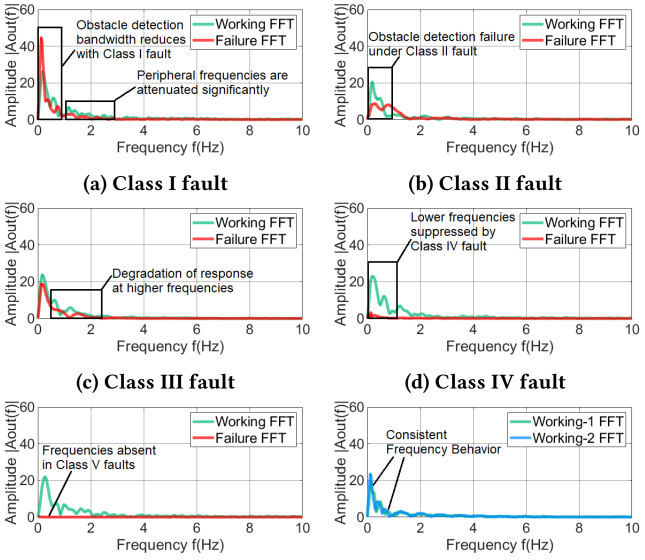
We co-located two sensors: *(i)* a tampered sensor (S_{tampered}) containing the failure and *(ii)* a working sensor (S_{working}) such that: *(a)* the distance between the sensors is closer than the size of the obstacle, *(b)* the obstacle is moved in a plane such that it comes into the detection region of both sensors simultaneously and *(c)* the obstacle is larger than the distance between the two sensors, allowing it to be incident on both the sensors simultaneously. Thus, we expect the same output from both S_{tampered} and S_{working}. We change the S_{tampered} sensor to test different types of failures described in §3.3.

We note the output signals (*i.e.,* A_{out} and C_{out}) in every failure scenario and compare it with a working sensor *to understand the impact of failure on the physics of the sensors.* Each experiment described below lasted for 15 minutes, when the obstacle (our palm in this case) was moved into the region of detection once every minute. We plot A_{out} for both a working sensor and every type of faulty sensor in **Figs.** 5b – 5f. The y-axis plots the voltage output of A_{out} and the x-axis is time. For a working sensor, we expect a spike in A_{out} once every 60 seconds denoting motion of the palm.

Lens dislocation (Class I) faults. The absence/dislocation of the lens causes imperfect focus of the thermal radiation resulting in the some residual output at the pyroelectric element even when there is no obstacle present. The impact of a Class I fault on A_{out} is shown in **Fig.** 5b. We can see that both the tampered and working sensors are still able to detect the obstacle (as shown by the 15 vertical spikes of A_{out}). However, when the obstacle is not present, we observe that the noise of A_{out} is much higher with a Class I fault as compared to a working sensor. Thus, it is important to look beyond just C_{out} signal to identify such failures.

Lens deformation (Class II) faults. Class II faults can potentially lead to missing the obstacles in the periphery of the field of view due to deformation or loss of material integrity (*e.g.,* puncture). **Fig.** 5c shows the A_{out} for a sensor with a Class II fault. The amplitude of A_{out} at the times of obstacle detection is lower when compared to a working sensor. This is expected since the damage to the lens leads to reduced thermal radiation incident on the pyroelectric subsystem. In general, Class II faults creates blind spots in the lens aperture that reduce the A_{out} amplitude.

Lens hindrance (Class III) faults are caused due to foreign particles entering the lens that compromise its optical capability, thereby affecting the intensity and angle of radiation incident on the pyroelectric element. We induce such faults by depositing some dust on the lens. **Fig.** 5d shows that the amplitude of A_{out} varies at each of the detection points. The variation depended on the amount



(a) **Class I fault**

(b) **Class II fault**

(c) **Class III fault**

(d) **Class IV fault**

(e) **Class V fault**

(f) **Comparing A_{out} of two working sensors**

Figure 6: Frequency Domain Representations of A_{out} under different faults in a PIR sensor. Only frequencies upto 10 Hz are shown because the human motion range most PIR sensors detect are in that range.

of dust with respect to the orientation of the obstacle. Though the obstacle was still being detected, we note that with increased dust deposition, the amplitude of A_{out} can fall below the comparator threshold required to capture the obstacle resulting in failure.

Failure in pyroelectric components (Class IV). Class IV faults happen when the optical filter on the pyroelectric element comes in contact with contaminants such as oil, mist or other smudge, it can lead to potentially missing obstacles. High temperatures can also cause expansion of the optical filter that can also result in failures. We induce Class IV faults by spraying some oil on the optical filter. We observe in **Fig** 5e that A_{out} has significantly attenuated spikes at each of the points that correspond to obstacle motion, but the spikes are not high enough to cause C_{out} to drop to LOW. This results in the obstacle to be missed completely.

Failure in electronics (Class V) faults. These are usually electronic faults such as short circuits or open circuits. They can cause the A_{out} or C_{out} values to be 'stuck' at certain anomalous value such as HIGH (3.3 V) or LOW (0 V). As a result, these failures can cause the obstacle to be completely missed as seen in **Fig.** 5f. The A_{out} waveform here is a flat, horizontal line and lacks oscillations.

Insights into underlying physics The core insight here is that A_{out} can shed light on the physical and electrical operating conditions of a PIR sensor as opposed to the mere boolean occupancy indicated by C_{out}. Thus, being able to snoop in on A_{out} can capture the interaction of IR radiation on the different sensor subsystems.

4.2 Frequency Domain Characterization

Till now we have showed A_{out} signal can be used to detect various failure scenarios. We transform time domain A_{out} signal into the frequency domain using Fast Fourier transform (FFT) to derive robust features. FFT offers high resolution in the frequency domain and deconstructs the frequencies and harmonics present in a signal.

We plot the FFT representations of A_{out} corresponding to different faults in **Figs.** 6a– 6e as compared to a working sensor, all under the presence of an obstacle. The y-axis plots the magnitude of FFT coefficients and the x-axis plots frequencies in the human range of motion (0 – 10 Hz). In case of a working sensor, we observe 2 big peaks and additional peripheral frequencies up to 4 Hz. The frequency spectrum depends on the type of faults.

Class I faults We observe in **Fig.** 6a that a dislodged lens cap (either partial or complete) leads to reduced information capture (lower bandwidth) and as a result, the sensitivity in the periphery of the sensors reduces. Note that there is just a single prominent peak in the faulty sensor (at a slightly lower frequency when compared to the normal sensor) that corresponds to the obstacle being perfectly aligned with the center of sensor as it passes. The magnitude reduces sharply as it moves away from the center of the sensor.

Class II faults Fig. 6b plots the FFT for a sensor with deformed lens cap. We note that Class II failures lead to significantly suppressed primary peaks (approx. 10 db). In addition, the peripheral frequencies are attenuated as observed in Class I.

Class III faults due to accumulation of foreign particles in the lens, lead to reduced sharpness of frequencies. We observe in **Fig.** 6c that the degradation starts with damping of the higher frequencies. As the amount of dust increases, we observed the frequencies getting increasingly damped until the object detection starts failing.

Class IV faults We observe in **Fig.** 6d that the frequencies present in the output are heavily suppressed resulting in missed obstacle detection. This is expected since the optical filter plays a major role in filtering out the non-human range frequencies from the thermal radiation and only passing through the frequencies corresponding to the human range. The presence of oil condensation or heat sinks causes absorption of the thermal radiation resulting in insufficient heat to cause pyroelectric effect.

Class V faults such as open and short circuits are easy to detect due to the absence of any prominent frequencies. Consequently, A_{out} is *stuck at* a value leading to only dc frequencies being present in the output. This result is observed in **Fig.** 6e, where the C_{out} and A_{out} pins are shorted on the board.

Summary: Frequencies present in FFT representations of A_{out} provide hints regarding status (*i.e.,* working/failed) of PIR sensors.

5 FAULT DETECTION AND DIAGNOSIS

In §4, we analyzed the impact of failures on the characteristics of the A_{out} output signal in both time domain and in frequency domain using FFT. In §5.1, we show that in every type of failure, there exists information about the failure and quantify the difference in physics captured by FFT coefficients using a K-S statistic test [16].

5.1 Learnability of the Analysis

We performed a 2-sample Kolmogorov–Smirnov (K-S) statistic test [16] to validate our hypothesis that the A_{out} distributions from both working and failed sensors are different. We performed the tests at a significance level (α) = 0.05, that results in a K-S statistic threshold of 0.1 – a standard threshold for checking if sample differences in two distributions imply difference in population. **Table** 2 shows the K-S statistic (kstest2 in Matlab [17]) computed for each class of failure relative to a working sensor in columns FC1 – FC5. We make the following observations :

Failed sensors have different frequency characteristics compared to a working sensor: Each failure class FC1 — FC5 has a

Table 2: **K-S Statistic Results: FC1-FC5 : failure class I-V. W : working sensor.**

Failure	FC1	FC2	FC3	FC4	FC5	W
K-S Test	0.41	0.33	0.49	0.65	0.95	0.09

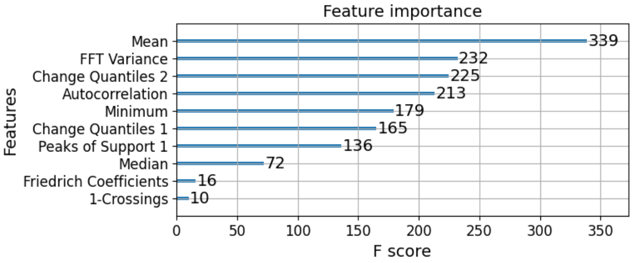


Figure 7: **Top 10 features selected using B-H procedure.**

K-S statistic value > 0.1 when compared to a working sensor (W), indicating that the distributions of faulty and working sensors dffer. **The worse the failure, higher the K-S statistic:** A sensor with oil condensation on the lens (C4) (a more pernicious fault leading to missed obstacles) resulted in a K-S statistic of 0.65 compared to 0.33 for a sensor with lens deformation (C2) (milder fault resulting in some blind spots). The K-S statistic computes the divergence between the physics of failed and working sensors.

Working sensors have similar characteristics: We performed a sanity check between distributions of multiple working sensors and noticed the consistency among them in the frequency domain. Computing the K-S statistic resulted in a value below the 0.1 threshold implying similar physics as seen in **Fig. 6f**.

5.2 Feature Selection and Importance

We use the time domain and frequency domain features of the A_{out} signal to classify the sensor to either working or one of the faulty classes. As the number of features that can be derived is huge (*e.g.*, in our case 305), we use the Benjamini-Hochberg (B-H) feature selection algorithm [3] to estimate and analyze *feature importance* in the A_{out} collected for working and each class of faulty sensors.

This technique is used offline with training data to decide which features are useful in predicting the label of the sensor (*e.g.*, working vs. class X). The output of B-H feature selection process is a feature importance score (F-score) that indicates how useful or valuable each feature is in the decision making process. The higher the F-score, the more important the feature is to the prediction.

B-H feature selection process works by initially training an ensemble of decision trees on all features, derived from both time and frequency domain features as mentioned in Christ *et al.* [7] and implemented in the open source library `tsfresh` [5]. It then measures the prediction accuracy using every feature. This gives a high accuracy at the expense of overfitting. Thereafter, each feature vector is independently evaluated with respect to its significance for prediction using hypothesis testing, assigning it a F-score. It then iteratively prunes features having low F-scores, trains decision trees using these reduced features and measures prediction accuracy. The process stops when a user-set threshold of accuracy is met or when all the combinations have been tested.

Applying the B-H process to our training data containing both faulty and working sensors, *we pruned 305 features for our entire data set to obtain 10 features at a slightly better accuracy.* **Fig. 7**. plots the F-score for the selected 10 key features in both time and frequency domains. These include: *(a)* FFT Variance that indicates how spread the frequency distribution is, *(b)* Autocorrelation that generalizes periodicity or patterns in the time domain, and *(c)* Measures of central tendency such as mean, median and change quantiles that measure the time domain changes within a corridor. This process is performed offline and the significance of the pruned feature set is guaranteed by the B-H algorithm.

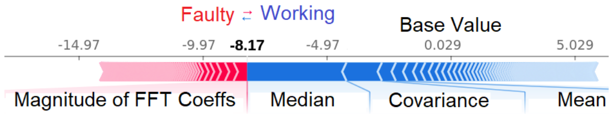


Figure 8: **Summary of Shapley Values showing FFT coefficients separate faulty and working classes.**

5.3 Monitoring Feature Performance using Shapley Additive Explanations (SHAP)

Once the important features are identified using the B-H feature selection process, we use the technique of SHAP [13] values to *explain or interpret* the output of a prediction. SHAP uses game-theoretic techniques to explain how the output of a prediction changes by conditioning on every feature present in the reduced feature set given by B-H process.

Fig. 8 shows an example of how SHAP values are analyzed for a particular prediction on a classification between faulty and working sensors. The features in red contribute to the sensor being predicted as faulty, whereas the features in blue contribute to the sensor being predicted as working. In this prediction, the FFT coefficients push the prediction to that of a faulty sensor while the mean, median and covariance push the prediction to a working sensor. The base value is the average SHAP value output by the model for the entire training dataset, and the value in bold is the SHAP value for this particular prediction. We leverage an open-source library implementation of the algorithm from Lundberg *et al.* [6, 13] that sorts the features by the sum of SHAP value magnitudes over all the samples and conditionally rejects the feature contributing the lowest to the prediction. Thus, we use the feature explanations from SHAP values to tune the model performance.

5.4 PIRMedic: Fault Detection & Analysis Algorithm

We now present our algorithm PIRMedic towards fault detection in PIR sensors. PIRMedic consists of 2 stages: *(i)* pre-deployment (training) stage, wherein the reference A_{out} values for a set of sensors (both working and faulty) are collected in an offline manner and analysed to extract key features and *(ii)* deployment stage, wherein the features from a operational A_{out} signal is matched with previously extracted features in order to detect and isolate faults. We next describe both the stages in detail.

5.4.1 Pre-deployment stage. We first collect A_{out} from a set of sensors, both working and faulty. The collected time series data are labeled to form a training dataset. We perform the Benjamini-Hochberg feature selection on this dataset to extract key features such as FFT coefficients of A_{out} for each class of sensor. This forms a feature dictionary containing a smaller number of features along with the failure as a label. We further verify the importance of these features using SHAP analysis and refine if required. We then build

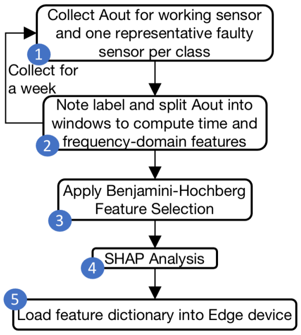

Collect Aout for working sensor and one representative faulty sensor per class
1
Collect for a week
Note label and split Aout into windows to compute time and frequency-domain features
2
Apply Benjamini-Hochberg Feature Selection
3
SHAP Analysis
4
Load feature dictionary into Edge device
5

Figure 9: Pre-deployment stage.

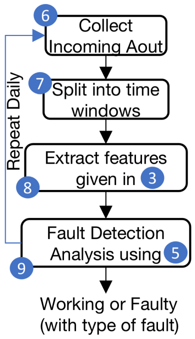

6
Collect Incoming Aout
Repeat Daily
7
Split into time windows
Extract features given in 3
8
Fault Detection Analysis using 5
9
Working or Faulty (with type of fault)

Figure 10: Deployment stage.

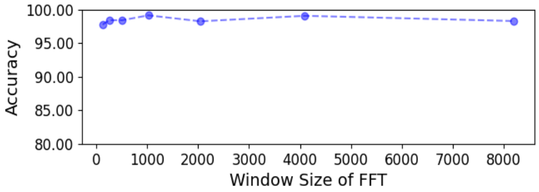


Figure 11: Model Accuracy for different window sizes. We choose 1024 as the default window size.

a classifier model to uniquely identify the failure in the sensor and load this model into the edge device connected to the PIR sensor.

We now describe the steps involved (**Fig. 9**): ① We deploy the working and faulty sensors for a short duration (say few days) in a real-world environment to collect realistic A_{out} signals for low, medium and high occupancy. Note this is a one-time activity performed for a specific PIR sensor type/manufacturer. We used 15 sensors comprising a mix of working and faulty sensors capturing failures in each class as our training set. ② We note the label of a sensor and split the A_{out} values into equal-sized time windows to calculate different time and frequency-domain features. ③ We apply B-H feature selection process for each type of sensor to identify unique and key features. ④ We performed SHAP analysis to understand the performance of each feature towards classification and refine them accordingly. ⑤ We use the final set of features along with class labels to build a classifier model and use random forest as our classifier model. This model is then loaded onto the edge devices for fault detection and analysis.

5.4.2 Deployment stage. This stage consists of the following steps (**Fig. 10**): ⑥ First, the operational A_{out} output signals for the deployed sensors are collected. ⑦ We split the A_{out} into equal-sized windows using the pre-deployment window sizes. ⑧ We extract the necessary features from the A_{out} signal. ⑨ We use a classifier to *(a)* isolate faulty from working sensors and *(b)* identify the class of failure in the faulty sensor if applicable.

As mentioned earlier, the pre-deployment stage (to collect features and build a classifier model) is a one-time activity. The fault detection (steps ⑥ - ⑨) depends on the application requirements, *i.e.,* every hour, day or week.

6 IMPLEMENTATION

Our solution is designed using commodity off the shelf (COTS) components. Our open sourced implementation[2] includes:

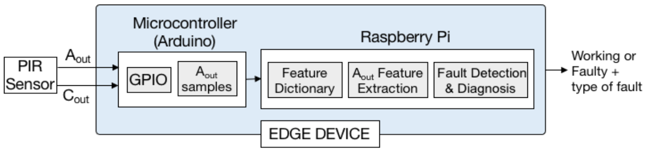

PIR Sensor
A_{out}
C_{out}
Microcontroller (Arduino)
GPIO
A_{out} samples
Raspberry Pi
Feature Dictionary
A_{out} Feature Extraction
Fault Detection & Diagnosis
EDGE DEVICE
Working or Faulty + type of fault

Figure 12: Workflow of PIRMedic for fault detection and diagnosis.

(a) **Edge Hardware** consisting of – *(i)* a PIR sensor [23], *(ii)* a bare-metal platform *viz.,* Arduino Mega Microcontroller Unit (MCU) for sensor data capture, and *(iii)* a linux-based platform *viz.,* Raspberry Pi for performing the fault detection and diagnosis.

The Arduino MCU polls the sensor hardware signals (*i.e.,* C_{out} and A_{out}) periodically over a GPIO interface. We poll the PIR sensors at a frequency ≥ 20 Hz due to the *Nyquist criterion* as the human motion information in the PIR sensor is between $0 - 10$ Hz. The Raspberry Pi platform performs the detection and analysis of faults for the data captured at the Arduino MCU.

(b) **Edge Implementation** We used standard python libraries such as `tsfresh` [5, 8] for FFT implementations to implement feature extraction, `numpy` for data preprocessing and `xgboost` [6] for implementing the machine learning classifiers. The end-to-end workflow of our solution is illustrated in **Fig. 12**.

6.1 Parameters of PIRMedic

(i) **Size of Window**: The collected A_{out} waveform is split into equal-size sample windows over which features are calculated. We observed that using a window too small (*e.g.,* 128) does not allow us to capture sufficient features for a failure type and can lead to a large number of false alarms. Also, a large window size (*e.g.,* 8192) leads to an overlap in the features of multiple failure classes leading to a loss of accuracy. The variation of model accuracy as a function of window size is shown in **Fig. 11**. We chose a window size of 1024 samples as it gave us a good tradeoff between time to capture a window (50 seconds) and accuracy (> 98%).

(ii) **Benjamini-Hochberg (B-H) Feature Selection** While there are more than 300 features that can be derived for time series analysis of A_{out}, B-H feature selection process (derived from parameterized hypothesis testing) [3, 7] prunes this to a lower number of features that can sufficiently capture the physics of the system. We observed that using more than 10 features did not contribute to a significant increase in accuracy (beyond 98%) and can, in practice, lead to lower performance due to overfitting as described in §5.2.

(iii) **Classifier Model**: We used a Random Forest classifier [24] owing to the capability to classify data based on "entropy" or the information gain. Random Forest uses an ensemble of trees and has been shown to give high predictive accuracy and control overfitting in practice, both of which are issues with decision trees [24]. Additionally, Random Forests are amenable to interpretation by techniques such as Shapley TreeExplainer that explains the features influencing the prediction. *Finally, we note that A_{out} is compatible with any classification algorithm.*

7 EVALUATION

The evaluation is intended to answer the following questions – *(a)* can we detect failures *i.e.,* separate faulty sensors from working sensors? and *(b)* what can we learn about the failure? In other words, can we perform diagnosis that can aid in replacing or repairing the

[2] Available at: https://github.com/synercys/PIRMedic

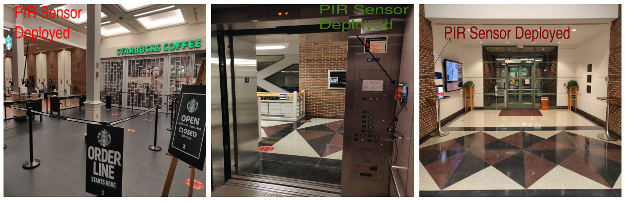


Figure 13: Deployment Scenarios used for evaluating our failure analysis.

sensor? We answer these questions in practical deployments, each consisting of the following stages:

(a) **Stage I: Failure Detection** The goal of this stage is to isolate defective, faulty sensors from functional, working sensors.

(b) **Stage II: Failure Diagnosis** In this stage, we diagnose the failure by mapping it to the taxonomy defined in §3.3. In other words, we seek to conclude whether it is the lens, pyroelectric or electronics subsystem that contains the failure.

(c) **Stage III: Fine-Grained Fault Analysis** In this stage, we learn details about the failure as precisely as possible. *E.g.,* if we identify a fault to be due to Class III failure, we seek to narrow it down to see if it is due to deposition of dust or paper on the lens.

7.1 Real-world Deployments

We performed the following *deployments across multiple scenarios* – *(a)* in the elevator of a building (§7.1.1), *(b)* in the lobby of a building (§7.1.2), *(c)* in a starbucks coffee shop (§7.1.3). The different deployment scenarios, shown in **Fig. 13**, capture diverse environmental conditions and challenges (heat, dust, humidity *etc.*). We used 15 sensors in total across all our deployments comprising of 5 working sensors and 2 sensors belonging to each faulty class. A combination of these was used in each of our deployments. These sensors were deployed (**Fig. 4**) over a duration of 3 months and collected data both during times of low occupancy (weekends, late nights) and high occupancy (weekdays, working hours). Note that no private data such as audio or video were captured during the deployments.

7.1.1 Deployment in a University Elevator (US). We deployed working and faulty sensors along the inside wall of an elevator in our university research building (middle **Fig. 13**). The data collection captured scenarios of both high and low occupancy during and outside of business hours.In addition to faulty sensors, we placed a working sensor to keep track of the actual motion in the elevator. **Fig. 14a** plots C_{out} and A_{out} values captured by normal sensors showing the actual traffic in the elevator on one of the days.The vertical stripes on the graph showing the oscillations of C_{out} (in blue) are points of occupancy *i.e.,* person entered/exited the elevator. Likewise, the regions of graph where there is no occupancy is represented by a flat line, hovering around 3.7 V.

For faulty sensors, we measure the number of time windows where an obstacle was captured or missed. The performance of faulty sensors relative to a working sensor is tabulated in **Fig. 14a**. When the lens cap is dislodged, it misses obstacles in 343 time windows whereas the (thermal) breakage of the pyroelectric element results in 247 misses. Overall, we observe that missed obstacles are more common than false alarms. This ties into common observations in conference rooms operated with PIR sensor-controlled lighting, where lights turn off even with occupants present.

As described in §5.4, PIRMedic collects the A_{out} signals and uses a machine learning model on their extracted features detect faulty sensors. Using a random forest classifier, we observe that the confusion matrix (**Fig. 14b**) shows that predicted label matches the true label for each failure. We note that – *(a)* the working sensor has distinct characteristics compared to the faulty sensors and is isolated correctly, *(b)* the faulty sensors with different failures in lens and pyroelectric subsystem are identified to the failure class correctly as observed by diagonal of the confusion matrix. Clearly, the features of A_{out}, derived by PIRMedic is able to differentiate between lens being dislodged, lens being covered with tape, oil condensation on the optical filter and the heat damage on the pyroelectric element.

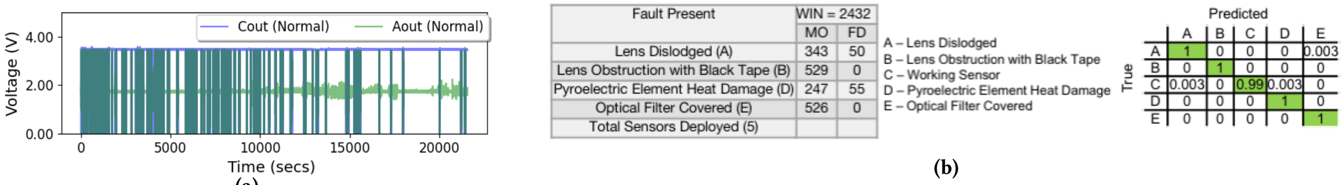
7.1.2 Deployment in a University Lobby (US). We deployed our sensors in the first floor lobby space of our university building. The lobby deployment is noisier with fewer constraints on direction of motion, compared to the confined space of an elevator.

Fig. 15a shows a 3 hour slice of the deployment. The vertical stripes on C_{out} correspond to traffic entering/exiting the lobby in the field of view. The performance of the sensors deployed, in terms of false detection and missed obstacles, is shown in the table in **Fig. 15b**. The sensors with lens dislodged missed a large amount of obstacles. This is due to the lack of focus of thermal radiation on the pyroelectric element resulting in dispersion of radiation due to the large lobby area. As a result of this failure, only obstacles that line up in the reduced field of view, are captured. (This issue is not present in an elevator due to its confined space.) Upon extracting the features of the A_{out} signals, and classifying it using the procedure in §5.4, we observe the confusion matrix as shown in **Fig. 15b**. The high fraction along the diagonal shows that a majority of sensor failures are classified and identified correctly. Note that we observe some misclassifications of sensor faults (*e.g.,* for C). We argue that this is due to the deployment area of a lobby being open and the limited coverage area of the sensor, both of which can lead to imperfect data at the periphery. Consequently, for traffic at the edges, the reduced thermal radiations can lead to some misclassifications.

7.1.3 Deployment in a Starbucks (US). We deployed our PIR sensors near the ordering queue of the coffee shop during business hours to capture the typical foot traffic (left **Fig. 13**). Our deployment captured linear, regularized bidirectional movement (entry followed by exit of the region of interest) (**Fig. 16a**). Our deployment lasted for a duration of 4 hours on a busy Friday evening. We observed that out of 226 time windows (each containing 1024 samples), there were missed obstacles across: *(i)* 27 windows in class I, *(ii)* 35 windows in class II, *(iii)* 26 windows in class III and no windows in class IV. The missed obstacles and false detections (none in this case) are summarized in **Fig. 16b**. Using PIRMedic, we observe high fractions along the diagonal of the confusion matrix; hence the predicted label of sensor failure matches the true label.

7.2 Stage I – Failure Detection

We aggregated all the sensors (both working and faulty) to *separate faulty sensors (without specifying which class) from working sensors.* We collected FFT coefficients of A_{out} on a pre-deployment lasting 1 week and performed daily collection of A_{out} signals from all the deployed sensors. We used a Random Forest classifier with specifications shown in **Fig 17**. Once predicted, we manually verified the

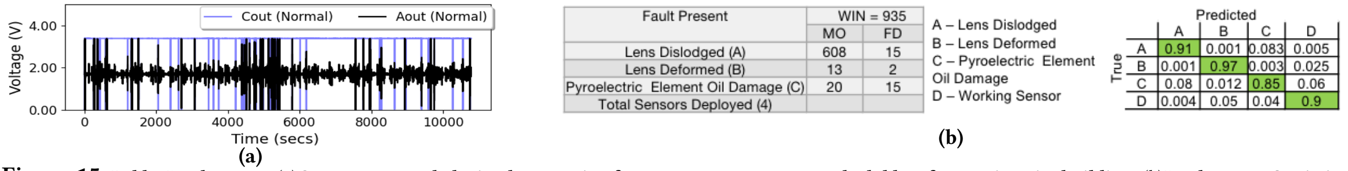


Fault Present	WIN = 2432	
	MO	FD
Lens Dislodged (A)	343	50
Lens Obstruction with Black Tape (B)	529	0
Pyroelectric Element Heat Damage (D)	247	55
Optical Filter Covered (E)	526	0
Total Sensors Deployed (5)		

A – Lens Dislodged
B – Lens Obstruction with Black Tape
C – Working Sensor
D – Pyroelectric Element Heat Damage
E – Optical Filter Covered

True \ Predicted	A	B	C	D	E
A	1	0	0	0	0.003
B	0	1	0	0	0
C	0.003	0	0.99	0.003	0
D	0	0	0	1	0
E	0	0	0	0	1

Figure 14: Elevator Deployment *(a)* Occupancy : The graph is a 6 hour deployment during business hours, *(b)* Deployment Statistics and confusion matrix of the classification model. Column titles: WIN→Total Number of Windows, MO→Misses Obstacle, FD→ False Detection.

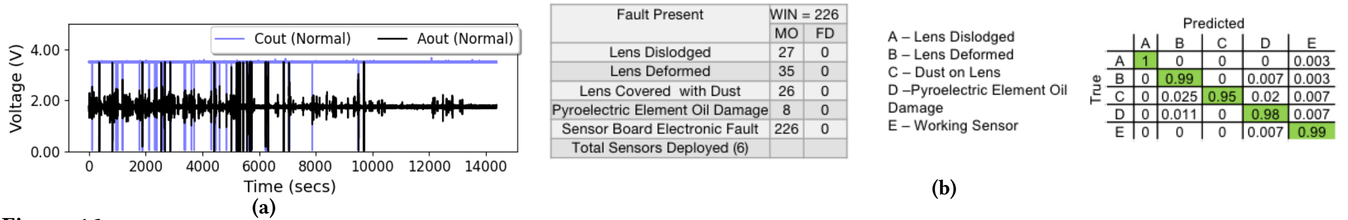


Fault Present	WIN = 935	
	MO	FD
Lens Dislodged (A)	608	15
Lens Deformed (B)	13	2
Pyroelectric Element Oil Damage (C)	20	15
Total Sensors Deployed (4)		

A – Lens Dislodged
B – Lens Deformed
C – Pyroelectric Element Oil Damage
D – Working Sensor

True \ Predicted	A	B	C	D
A	0.91	0.001	0.083	0.005
B	0.001	0.97	0.003	0.025
C	0.08	0.012	0.85	0.06
D	0.004	0.05	0.04	0.9

Figure 15: Lobby Deployment *(a)* Occupancy graph during late evening from 6.45 pm to 9.45 pm at the lobby of our university building, *(b)* Deployment Statistics and confusion matrix of the classification model. Column titles: WIN→Total Number of Windows, MO→Misses Obstacle, FD→ False Detection.



Fault Present	WIN = 226	
	MO	FD
Lens Dislodged	27	0
Lens Deformed	35	0
Lens Covered with Dust	26	0
Pyroelectric Element Oil Damage	8	0
Sensor Board Electronic Fault	226	0
Total Sensors Deployed (6)		

A – Lens Dislodged
B – Lens Deformed
C – Dust on Lens
D –Pyroelectric Element Oil Damage
E – Working Sensor

True \ Predicted	A	B	C	D	E
A	1	0	0	0	0.003
B	0	0.99	0	0.007	0.003
C	0	0.025	0.95	0.02	0.007
D	0	0.011	0	0.98	0.007
E	0	0	0	0.007	0.99

Figure 16: Starbucks Deployment Occupancy: Evening deployment from 4.45 pm to 8.45 pm. The second half has much lesser little foot traffic. *(b)* Deployment Statistics and confusion matrix of the classification model. Column titles: WIN→Total Number of Windows, MO→Misses Obstacle, FD→ False Detection.

true status of the sensor and observed an accuracy of over 99% in predicting if the sensor is working or faulty. Thus, FFT-based features of A_{out} were able to capture the physics, and hence separate faulty sensor physics from working sensor physics.

7.3 Stage II – Failure Diagnosis

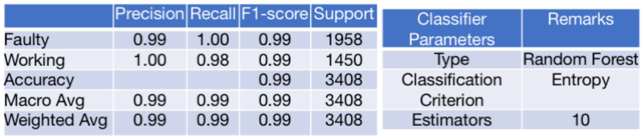
Diagnosing the *cause of failures (i.e., class)* requires further investigation of A_{out} from faulty sensors. **Fig. 18a** plots the CDF of FFT coefficients obtained for working sensors and different faulty sensors. The figure shows that although the distribution of working sensors (bottom-most curve) is distinct from failed sensors, different classes of failures have distributions that are not easily separable. Thus, relying on merely the FFT is insufficient. Consequently, we look at an additional 10 features as described in §5.2 — §5.3 and shown in **Fig. 7**. Failure diagnosis leverages the combined feature space to identify failure classes according to the taxonomy in §3.3.

We validated this across the multiple deployments in the wild — Elevator (§7.1.1), Lobby (§7.1.2) and Starbucks (§7.1.3).

Note: Our real-world measurements demonstrate that features of A_{out} extracted during training and B-H feature selection is robust to variations in shapes and sizes of obstacles.

7.4 Stage III – Fine-Grained Failure Analysis

In fine-grained fault analysis, we point to the *precise reason* for the fault where applicable. For example, in Class III faults it is useful to distinguish between different foreign substances contaminating the lens that can cause poor performance.



	Precision	Recall	F1-score	Support
Faulty	0.99	1.00	0.99	1958
Working	1.00	0.98	0.99	1450
Accuracy			0.99	3408
Macro Avg	0.99	0.99	0.99	3408
Weighted Avg	0.99	0.99	0.99	3408

Classifier Parameters	Remarks
Type	Random Forest
Classification Criterion	Entropy
Estimators	10

Figure 17: Fault Detection showing that faulty sensors isolated from working sensors by extracting FFT of A_{out} and using a Random Forest Classifier.

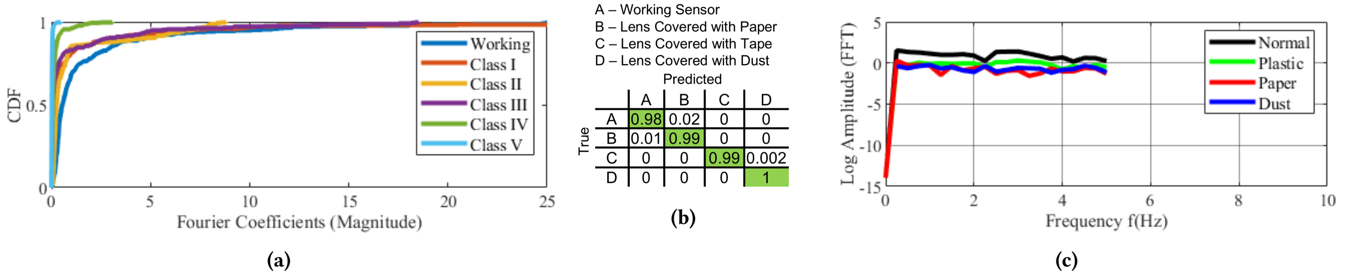
To investigate this, we selected the set of faulty sensors identified as Class III faults in stage II: some covered with paper, some with plastic tape and some with dust and observe the A_{out} output closely. Although all the failure scenarios result in the obstacle being missed, frequencies present in A_{out} are different in each case (**Fig. 18c**), due to the distinct thermal absorption characteristics of the materials. Hence, we observe that the FFT coefficients can separate Class III failures into different fine-grained sub-classes as shown in **Fig. 18b**.

It is to be noted that fine-grained fault analysis does not work in all cases. For example, differentiating between a lens puncture (a hole in the lens) and a lens deformation (defect in lens curvature), both examples of Class II failures, is complex. This is because both lens puncture and lens deformations lead to blind spots in the frequency spectrum. Thus, we need to use knowledge of the subsystem physics before applying the fine-grained analysis.

Summary: *(i)* The FFT representation of A_{out} is used to perform binary classification between working and faulty sensors, *(ii)* Considering multiple direct and derived time and frequency domain signal characteristics of A_{out} can diagnose the cause of failure in the sensor and *(iii)* Fine-grained fault analysis is possible in select failure cases (*e.g.,* Class III) by reanalyzing the FFT waveforms.

8 DISCUSSION

Practicality: A_{out} is an intrinsic signal that is present in all PIR sensors. Commercial sensors typically do not provide easy access to the A_{out} signal. While there exists some commercial sensors (*e.g.,* [21, 22]) that expose A_{out}, we believe that our work can provide further incentive to other manufacturers. In addition, existing sensors can be modified to expose A_{out}— most do as a test point on



	A	B	C	D
A	0.98	0.02	0	0
B	0.01	0.99	0	0
C	0	0	0.99	0.002
D	0	0	0	1

Figure 18: (a) CDF of FFT Coefficients showing different distributions depending on the class of faults.,(b) Fine-grained Fault Analysis,(c) A_{out} waveforms showing different behavior for paper, plastic and dust obstructions.

the PCB. This modification would not incur any additional hardware costs, with the exception of a new pin connector.

PIRMedic can precisely determine if a sensor is working or faulty in single time window comprising of 1024 samples. Since we sample at 20 Hz, the end-to-end process of data acquisition, processing and inference can be performed in a little over a minute.

Consistency of behavior: We observed that the inherent characteristics of the A_{out} signal, both in time and frequency domains, remain consistent across different manufacturers. This is expected since all PIR sensors use a pyroelectric element.

Design: We envision PIRMedic to be connected to a Building Monitoring System (BMS) and provide real-time fault detection. We intend to develop APIs for application developers to build higher-level diagnostic tools on top of PIRMedic.

Acknowledgments. We would like to thank Denizhan Kara (UIUC) and Kyo Kim (UIUC) for their insights on feature analysis. Further, we are grateful to Kapil Vaidya (MIT), Aditi Partap (Stanford), Hsuan-Chi Kuo (UIUC), Bin-Chou Kao (UIUC) and Ragini Gupta (UIUC) for their detailed feedback.

REFERENCES

[1] Application Note Murata PIR Sensor. https://www.murata.com/~/media/webrenewal/products/sensor/infrared/applinote_pir.ashx?la=ja-jp.

[2] H. Ayadi, A. Zouinkhi, B. Boussaid, and M. N. Abdelkrim. A machine learning methods: Outlier detection in wsn. In *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 722–727, 2015.

[3] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[4] Tusher Chakraborty, Akshay Uttama Nambi, Ranveer Chandra, Rahul Sharma, Manohar Swaminathan, Zerina Kapetanovic, and Jonathan Appavoo. Fall-curve: A novel primitive for iot fault detection and isolation. In *Proceedings of the 16th ACM SenSys*, page 95–107, 2018.

[5] Maximilian Christ. tsfresh. https://github.com/blue-yonder/tsfresh, 2018.

[6] Maximilian Christ. Shap xgboost implementation. https://github.com/slundberg/shap, 2020.

[7] Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W Kempa-Liehr. Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package). *Neurocomputing*, 307:72–77, 2018.

[8] Maximilian Christ, Andreas W Kempa-Liehr, and Michael Feindt. Distributed and parallel time series feature extraction for industrial big data applications. *arXiv preprint arXiv:1610.07717*, 2016.

[9] C. Dorffer, M. Puigt, G. Delmaire, and G. Roussel. Blind mobile sensor calibration using an informed nonnegative matrix factorization with a relaxed rendezvous model. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2941–2945, 2016.

[10] Hubbell. Ceiling Hard Wired Occupancy Sensor, 2,000 sq ft Passive Infrared, Ultrasonic, Office White. https://www.hubbell.com/hubbell/en/Products/Electrical-Electronic/Wiring-Devices/Lighting-Controls/CeilingWall-Sensors/ATD2000CRP/p/1545054?location=&brandCode=WDK, 2020.

[11] Hubbell. Wall Hard Wired Occupancy Sensor, 1,600 sq ft Passive Infrared, Ultrasonic, Office White. https://www.hubbell.com/wiringdevice-kellems/en/Products/Electrical-Electronic/Wiring-Devices/Lighting-Controls/CeilingWall-Sensors/ATD1600WRP/p/1545050, 2020.

[12] Nodira Khoussainova, Magdalena Balazinska, and Dan Suciu. Towards correcting input data errors probabilistically using integrity constraints. In *Proceedings of the 5th ACM International Workshop on Data Engineering for Wireless and Mobile Access*, MobiDE '06, page 43–50, New York, NY, USA, 2006. Association for Computing Machinery.

[13] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 4765–4774. Curran Associates, Inc., 2017.

[14] B. Maag, Z. Zhou, and L. Thiele. A survey on sensor calibration in air pollution monitoring deployments. *IEEE Internet of Things Journal*, 5(6):4857–4870, 2018.

[15] Sumukh Marathe, Akshay Nambi, Manohar Swaminathan, and Ronak Sutaria. Currentsense: A novel approach for fault and drift detection in environmental iot sensors. In *ACM/IEEE IoTDI*, 2021.

[16] Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[17] MATLAB Documentation Mathworks. kstest2: Two sample Kolmogorov-Smirnov test. https://www.mathworks.com/help/stats/kstest2.html, 2015.

[18] J. Murphree. Machine learning anomaly detection in large systems. In *2016 IEEE AUTOTESTCON*, pages 1–9, 2016.

[19] Sujay Narayana, R. Venkatesha Prasad, Vijay S. Rao, T. V. Prabhakar, Sripad S. Kowshik, and Madhuri Sheethala Iyer. Pir sensors: Characterization and novel localization technique. In *Proceedings of the 14th ACM IPSN*, page 142–153, 2015.

[20] PIR Fresnel Lens, Infrared Fresnel Lens. https://3dlens.com/pir-fresnel-lens.php.

[21] Panasonic. AMN22111 Industrial Devices. https://bit.ly/3Al6lvU, 2021.

[22] Panasonic. AMN23111 Industrial Devices. https://bit.ly/2VSawjM, 2021.

[23] PIR Sensor Evaluation Board (IMX-070, IMX-060), Murata. https://bit.ly/2Y8Dj51.

[24] Robi Polikar. Ensemble learning. In *Ensemble machine learning*, pages 1–34. Springer, 2012.

[25] Alexander Power and Gerald Kotonya. Providing fault tolerance via complex event processing and machine learning for iot systems. In *Proceedings of the 9th International Conference on the Internet of Things*, IoT 2019, New York, NY, USA, 2019. Association for Computing Machinery.

[26] Haroon Rashid, Pandarasamy Arjunan, Pushpendra Singh, and Amarjeet Singh. Collect, compare, and score: a generic data-driven anomaly detection method for buildings. In *Proceedings of the Seventh International Conference on Future Energy Systems Poster Sessions*, pages 1–2, 2016.

[27] Olga Saukh, David Hasenfratz, and Lothar Thiele. Reducing multi-hop calibration errors in large-scale mobile sensor networks. In *Proceedings of the 14th ACM IPSN*, page 274–285, 2015.

[28] Abhishek B. Sharma, Leana Golubchik, and Ramesh Govindan. Sensor faults: Detection methods and prevalence in real-world datasets. *ACM Trans. Sen. Netw.*, 6(3), June 2010.

[29] Arjun Tambe, Akshay Nambi, and Sumukh Marathe. Is your smoke detector working properly? robust fault tolerance approaches for smoke detectors. In *Proceedings of the 19th ACM MobiSys*, page 310–322, 2021.

[30] Guoming Tang, Zhen Ling, Fengyong Li, Daquan Tang, and Jiuyang Tang. Occupancy-aided energy disaggregation. *Computer Networks*, 117:42–51, 2017.

[31] Y. Wang, A. Yang, Z. Li, X. Chen, P. Wang, and H. Yang. Blind drift calibration of sensor networks using sparse bayesian learning. *IEEE Sensors Journal*, 16(16):6249–6260, 2016.

[32] Kamin Whitehouse and David Culler. Calibration as parameter estimation in sensor networks. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 59–67, 2002.

[33] W. Wu, X. Cheng, M. Ding, K. Xing, F. Liu, and P. Deng. Localized outlying and boundary data detection in sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 19(8):1145–1157, 2007.

[34] Yun Xiang, Lan Bai, Ricardo Piedrahita, Robert P. Dick, Qin Lv, Michael Hannigan, and Li Shang. Collaborative calibration and sensor placement for mobile sensor networks. In *Proceedings of the 11th ACM IPSN*, page 73–84, 2012.

[35] Ke Yan and David Zhang. Improving the transfer ability of prediction models for electronic noses. *Sensors and Actuators B: Chemical*, 220, 12 2015.