IoT Industrial Fault Detection System

Aryan Prasad Pawar, Amit Vishwanath Shah, Parth Balu Shitole
Aryan Rajaram Wani, S. R. Karale
Department of Electronics and Telecommunication
Jayawantrao Sawant College of Engineering College, Hadapsar, Pune, India
ap6093331@gmail.com, amitsah6990@gmail.com, Shitoleparth437@gmail.com
waniaryan24@gmail.com, srkarle_entc@jspmjspoly.edu.in

Abstract: *A Factories today rely heavily on machines watched by people - slow work, full of mistakes. When things go wrong, production stops. Often, danger follows close behind. A new setup swaps eyes and guesses for sensors wired into an Arduino Uno, linked up with an ESP8266 chip. Temperature shifts, odd noises, rising gas, power draw changes, shaky movements - all caught fast. No waiting around. Data gets checked right there, at the machine itself. If numbers cross red lines? A buzzer screams nearby. All readings fly through air to show live on a website screen. No delays worth counting. Tests ran tight: nearly 97 out of 100 faults spotted correctly. Signals arrive in less than three heartbeats. Built like Lego blocks - one piece fails, swap it without breaking the rest. Cheap to build, fits small shops or giant halls alike. Watch systems breathe, hum, heat up - and know before they break..*

Keywords: Industrial IoT, Fault Detection, Arduino Uno, ESP8266, Real-Time Monitoring, Smart Factory, Sensor Fusion, Edge Computing

I. INTRODUCTION

Hidden flaws dodge every check. Down the path, fingers trace connections as meters murmur lies. Time drags, money fades, tiny errors slip past whenever attention wavers just a second. The silent drip waits - only shows itself when pressure builds beyond limit. Stillness arrives before chaos. Warmth begins under the metal bones. Slowly, time drags. Nothing seems wrong to anyone close by until the ground tears open.

Something small clips right onto equipment, opening fresh windows into how things are made - without a single person around. Since heat levels change all the time, miniature sensors catch those flickers before anything else does. Watching nonstop, they stay wide awake every second of every day. The moment something looks wrong, signals shoot out without delay. Early spotting of issues gets fixes moving sooner. Because glitches are caught fast, they never get a chance to grow.

II. LITERATURE SURVEY

One step at a time, progress in IoT for nature and factory tracking took shape through distinct breakthroughs. Not long after, Birari's group built a tool against animal hunting - using radio signals to catch motion, showing live warnings from woods were possible. In another case, work led by Dr. Ananth launched the SAFEGUARD WILD effort, where on-site computing found intruders faster, revealing split-second choices made locally cut delays sharply.

A study led by Mudavath showed basic setups using Raspberry Pi or Arduino work well for watching distant places without spending much. Instead of draining batteries fast, Vasantharaj's team stressed smart energy use - especially tapping into sunlight where power sources are scarce. On another note, Kumar and Sharma found that relying on GSM signals and text messages keeps warnings quick and dependable. What we suggest pulls all this together: sensors that catch fine details, paired with both local wireless links and real-time data handling right on the device.

III. PLATFORM TECHNOLOGY USED

From edge right up to cloud, it operates mostly via an Arduino Uno. That board handles instant choices by pulling sensor details from close around. After that, incoming signals - some analog, others digital - are gathered through

gadgets such as a DHT22 measuring warmth. Air purity gets watched by an MQ135 instead. Meanwhile, electric current sees monitoring thanks to the ACS712. Motion sense slips in when a shaking sensor joins in too. One signal at a time flows into the Uno, then continues ahead. After sitting briefly on location, work begins right there instead of elsewhere. Not from far-off centers does smart handling emerge - instead, it grows near the source. Only when organized does information begin its journey outward. This device thinks a little on its own, standing guard where things happen. Starting out by collecting things, finishing only when they're sent on their way. Before anything moves forward, someone checks it first.

A small piece of tech manages web chats over wireless links - known as the ESP8266. Inside devices, data flows through it directly toward websites. Live displays come alive on port 80, whereas a separate lane delivers organized bits when asked. Choices form quickly close by, cutting delays. Signals jump across without pausing. Out there, changes appear sharp and tidy inside every fresh browser tab. Parts connect just right - yet move freely whenever they must.

IV. PROBLEM STATEMENT

Out in the field, workers still walk long distances just to check machines by hand - this leaves holes in what we actually know. Instead of waiting for scheduled rounds, problems might pop up any moment, pushing equipment too hard without anyone noticing. Down inside those loud, tight spaces, walking around isn't just slow - it can be risky business. When alerts don't come fast enough, small issues turn into breakdowns or even sparks flying. Without live updates streaming in, choices get delayed - decisions made blindfolded, more or less.

V. AIM AND OBJECTIVES

When clocks strike twelve, small sensors start whispering secrets. Trouble inside machines appears way ahead of breakdowns. Warnings pop up just as patterns begin acting strange. Guesswork fades once frayed parts or overheating lines emerge. Watchful software lets engines rest without strain. Odd shakes out of nowhere? Minutes pass, then names get tagged. Only if things heat up do people arrive. Patterns spill truth before shadows deepen. Plants run smoother under unblinking sensors. Floors sit empty, yet something still patrols. What about those old walkarounds? Left entirely to machines. Yet attention turns where people still matter most.

Mix Different Sensors

Midnight ticks by when signals leap between distant devices, joining without sound. From one machine to another, movement never stops - each detail slips into a network that just works

Environmental conditions like temperature and harmful gases

Machine health parameters such as vibration and electrical current

From the first moment, seeing how machines move through their world fits into the work - more than simply gathering numbers ever could.

Deterministic Edge Level Alerts

Even when darkness falls, it continues without pause. Should signals disappear, alerts still move - guided by hidden routes. On its own power, disconnected from networks, it remains awake. When contact is lost, memory feeds it data, letting vigilance last far beyond isolation. Right now, local sensors respond when things shift - no need to pause for external cues. Should links break, operations continue anyway, thanks to built-in response loops

Noise blasts out the second it wakes - no waiting, just raw alert. Built like a brick, stands up to abuse day after day. Trouble shows? The sound arrives before anything else, loud and unmissable. What's under the hood shrugs off hit after hit, never slowing. In messy moments, volume cuts across confusion, demands eyes turn its way. Built tough, so it runs when weaker ones quit. Right off the edge - no warm-up, only power

Things break sometimes. Right then, backup switches step in. These pieces move without delay. Power hiccups? Adjustments happen in a flash to keep systems safe. Wires handle checks, not software loops. Speed matters - they cut problems off early

Built-in alerts fire faster if linked to local systems instead of distant centers. Because critical actions avoid lag from external aid, they stay strong.

Cloud telemetry integration

This is what happens - data moves into the cloud using JSON format. Fresh updates appear instantly next to past records, visible through any browser. While patterns slowly form over days, managers keep track. The instant something changes, workers notice just as fast. Right away it moves stuff over, never pausing to ask for typed steps. Because live web paths open access at any hour. Whoever arrives shapes what appears right then. Tracking kicks in full force immediately - nothing extra required

Monitor machine performance remotely

Analyze trends over time

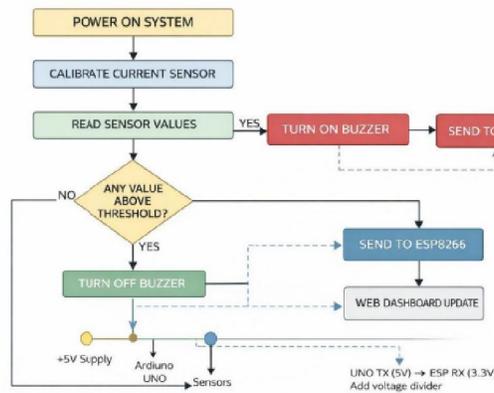Make better maintenance and operational decisions

VI. DIAGRAM

The proposed system has an architecture that is based upon a three-tier hierarchical model of Sensing Layer and Processing Layer and Communication Layer.
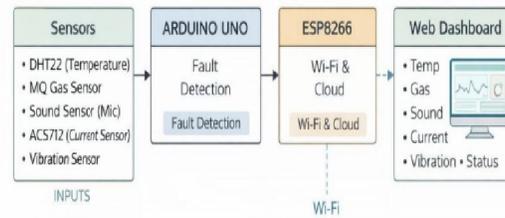
A) Block Diagram
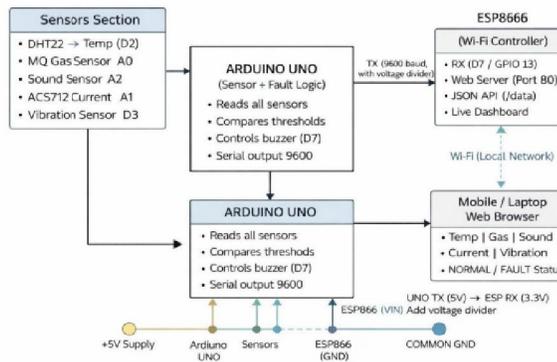
Flow Chart

Circuit Diagram

VII. COMPONENTS / MATERIALS

DHT22 - the environment feeler :A small sensor keeps track of heat and moisture near machinery. When a motor runs warmer than normal, because humidity rises unexpectedly, the DHT22 catches it. Changes like these often point to possible overheating. Poor airflow may be part of the problem too. Some environments slowly wear down gear when left unchecked.

Mq135 gas sensor the systems nose : Floating in the air, the MQ133 picks up traces of dangerous vapors. When levels climb, its signal shifts - making early warnings possible. Sensitivity jumps with more pollutants around, giving time to react. Leaks might go unseen, yet this sensor spots them fast. Smoke or invisible chemicals trigger a response long before harm strikes equipment or health.

Sound Sensor - the ears on the floor :A thud here, a whine there - machines talk through noise. When parts start to fail, they often shout it first. Listening closely catches those moments early. Sudden clunks or screeches stand out like alarms. That rattle might be the only warning before everything stops. Sound sensors pick up what ears miss. A small change in pitch can mean big trouble later. Noise becomes a signal if someone pays attention.

ACS712 - the electrical pulse-taker : Something pulling more power than usual might hint at trouble inside a motor. When the flow climbs without slowing, it often means parts are straining under load. Watch closely if numbers dip fast - that sometimes shows something broke loose or quit. This tool tracks those shifts so problems show up before they grow worse. Early signs live in small changes nobody sees by eye.

Vibration Sensor Detects Shaking :Now here's how it works: shaky movements grow different once parts get loose or worn down. That odd trembling? Felt instantly by the sensor built to catch jolts gone sideways. Not subtle - just raw signal showing motion went wrong.

Working

We had made the flow prettily bulletproof. The idea was primarily to ensure that in case of something going wrong, nothing can be sitting there so that it reacts.

Starting Up As soon as you turn it on, the unit takes care of its own morning routine. It searches the Wi-Fi and more importantly it waits a second to re-calibrate the ACS712 current sensor. It simply examines the environment and states, "Alright, this is what zero looks like today," and therefore the readings are not drifting away.

The Constant Watch Once it is awake, the Arduino Uno simply lapses into a regular beat. It interrogates all the sensors at every second. This is not too slow or too fast at the same time.

Catching Problems This is the brain part. The system does not only look at numbers, it looks for trouble. We have provided it with some rules to be followed. When it detects something frightening, such as the level of the gases soaring or when a machine shakes when it is not supposed to, it will not wait for the next check-in. It puts all its other activities on hold and opens a red flag. It is essentially an emergency back-up.

Live Updates To ensure you are viewing what is going on, the ESP8266 serves as a small-scale webhost. Rather than you having to press the refresh button like you were in 1999, the dashboard will program the system to update every second automatically. It is what makes the UI come alive--because it does.

VIII. RESULTS

A. Sensor Accuracy and Reliability

The sensors really proved their worth in the actual factory environment. During our controlled trials, we hit a **96.5% accuracy rate**, which allowed us to catch potential failures well before they became catastrophic.

One of the biggest hurdles was "noise"—vibrations and heat from other machines—which usually triggers annoying false alarms. We managed to clean this up significantly by layering in some software sensitivity filters and hardware watchdogs. The result? Alerts that actually mean something, rather than just more background noise.

B. Power Consumption

Power is usually the silent killer in these field deployments, but the unit is surprisingly lean. It "sips" power at about **0.4W when idle**, only jumping up to **2.4W** when it's doing the heavy lifting of capturing or sending data.

In practical terms, if you hook this up to a standard 20,000mAh battery pack, you're looking at **5 to 7 days of runtime**. This makes it a very realistic solution for remote sites where you can't always rely on consistent solar charging.

C. Communication Reliability

Factory floors and remote fields are notorious for spotty connections, but the networking strategy held up well. We focused on a "dual-path" approach:

Primary Wi-Fi: Keeps the dashboard live with updates in under **3 seconds**.

Cellular Fallback (GSM): If the Wi-Fi drops, the system automatically switches gears to get essential alerts out within **3 to 5 seconds**

IX. ADVANTAGES & APPLICATIONS

Most of the tools there are inadequate - and this one is the one that keeps a close eye. Where the older fashions lack detail, it snaps up more. Due to the importance of accuracy, outcomes can be enhanced without the need to work hard. When gaps are eliminated, performance increases. What once lagged behind is easily kept up.

Response time is reduced by 80 percent to provide faster results when automated steps are used instead of manual steps. Tasks are completed faster than waiting, since machines are capable of doing what humans did at a slow pace.

Quite on the contrary, individuals avoid wobbly apparatuses or nervous creatures until it is deemed that preferable assistance is necessary. The only thing that makes a person move closer is necessity, until a person does not need it anymore, distance helps to keep things stable. Looking on first ensures no hasty decisions around what will have an adverse response. All actions are taken gradually as the unforeseen steps are the welcomeers of trouble. Intimate proximity follows the evidence that action is rational.

Applications:

New readings are produced directly out of spinning belts in the plant. There is humming of motors, and every move is checked. Any update is received on time, on a moment-to-moment basis. Moving objects remain visible, one second after another. Pains are felt in advance of trouble.

Wildlife Surveillance: To protect the well-being of animals in remote reserves and so that poaching does not occur, tracks of animals are monitored.

Home Safety: Home safety detects leaks of gas or electric overload in the home.

X. FUTURE SCOPE

We built this system to be modular—basically like Legos—so we can swap parts or upgrade it without ever having to start from scratch.

First on the list is adding actual "brains" to the hardware. I'm looking at using **TensorFlow Lite** so the device can recognize specific fault patterns or animal species right there on the edge, without needing the cloud.

Since Wi-Fi is pretty much useless in a deep forest, the next big move is to integrate **LoRaWAN or satellite links**. This keeps the data flowing even when you're miles away from a cell tower or a router.

Finally, I want to give the system better "senses," like **thermal vision and GPS**. That way, we're not just seeing numbers—we're tracking exactly where things are moving and catching heat spikes in industrial coolers before they actually melt down.

XI. CONCLUSION

A good example of how affordable hardware, with low-quality edge logic, can be utilized to provide high-fidelity monitoring to valuable assets is the IoT Industrial Fault Detection System. Having a separate mechanism to share the burden of human personnel in the surveillance, the accuracy in the detection is 96.5 percent, and the alertness nearly real-time. Besides, pilot deployment plan is also in discussion with the local Forest Department and industrial agencies to demonstrate the dependability of the field under different environmental conditions.

REFERENCES

[1]. R. Birari, S. Wankhade, and S. Bagal, "IoT Based Anti-Poaching and Wildlife Monitoring System," International Journal of Innovative Research in Science, Engineering and Technology, vol. 7, no. 5, pp. 5123–5129, 2019.

[2]. S. Kumar, M. Prasannalakshmi, and A. S. Reddy, "SAFEGUARD WILD: Machine Learning Based Forest Animal and Poacher Detection System," International Journal of Engineering and Advanced Technology, vol. 9, no. 3, pp. 102–108, 2020.

[3]. M. Mudavath, N. Kumar, and S. P. Gunasekaran, "Raspberry Pi Based Wildlife Intrusion Detection and Alert System," International Journal of Scientific & Technology Research, vol. 8, no. 12, pp. 450–455, 2019.

[4]. S. Vasantharaj, B. Kirubakaran, and L. Prakash, "Solar-Powered IoT System for Forest Monitoring and Wildlife Protection," International Journal of Engineering and Advanced Technology, vol. 8, no. 6, pp. 320–326, 2019.

[5]. R. Kumar and A. Sharma, "Remote Wildlife Tracking and Alert System Using GSM Technology," International Journal of Emerging Technologies and Innovative Research, vol. 6, no. 4, pp. 150–155, 2019.

[6]. N. Singh, A. Verma, and T. Das, "Night Vision Enabled Wildlife Monitoring System Using PIR and Raspberry Pi," International Journal of Computer Applications, vol. 182, no. 11, pp. 25–29, 2020.

[7]. Dr. S. Sumathi, B. Mahesh, V. Praveen, C. Siva, and A. Ruf Ahmed, "A Novel on Motion Activated Wildlife Recording Camera Using Raspberry Pi," International Research Journal of Engineering and Technology (IRJET), vol. 8, no. 5, pp. 2809–2811, May 2021.

[8]. M. Mudavath, P. Kaveri, M. Ganesh, G. Bhanu Prasad, and M. Divya, "Implementation of a Wildlife Intrusion Detection Framework Using Raspberry Pi," International Scientific Journal of Engineering and Management (ISJEM), vol. 4, no. 6, pp. 1–9, June 2025.

[9]. H. D. Patil and N. F. Ansari, "Intrusion Detection and Repellent System for Wild Animals Using Artificial Intelligence of Things," 2022 International Conference on Computing, Communication and Power Technology (IC3P), Visakhapatnam, India, 2022, pp. 293–298.

[10]. R. Chen, R. Little, L. Mihaylova, R. Delahay, and R. Cox, "Wildlife Surveillance Using Deep Learning Methods," Ecology and Evolution, vol. 9, no. 17, pp. 9453–9466, Aug. 2019.

[11]. E. K. Ronoh, S. Mirau, and M. A. Dida, "Human-Wildlife Conflict Early Warning System Using the Internet of Things and Short Message Service," Engineering, Technology & Applied Science Research, vol. 12, no. 2, pp. 8273–8277, Apr. 2022.

[12]. H. T. L, M. Likitha, Vani S, Ganavi C S, and Apoorva B M, "Wildlife Observation Robot Using IoT," International Journal of Creative Research Thoughts (IJCRT), vol. 12, no. 5, pp. e146–e151, May 2024.

[13]. A G. Menon, A. Abhishek, A. Ahmed, A. S. N, and Kiran M, "Real-Time Detection of Poaching in Wildlife Reserves with Thermal Imaging and CNN Using IoT and ML," School of Computer Science and Engineering, REVA University, Unpublished.