Machine Learning Sensors: A Design Paradigm for the Future of Intelligent Sensors

Article *in* Communications of the ACM · October 2023

DOI: 10.1145/3586991

CITATION
1

READS
48

5 authors, including:

Matthew Stewart
Harvard University
24 PUBLICATIONS **138** CITATIONS
SEE PROFILE

Brian Plancher
Columbia University
36 PUBLICATIONS **409** CITATIONS
SEE PROFILE

Sachin Katti
Stanford University
144 PUBLICATIONS **21,441** CITATIONS
SEE PROFILE

Vijay Janapa Reddi
University of Texas at Austin
224 PUBLICATIONS **10,782** CITATIONS
SEE PROFILE

DOI:10.1145/3586991

Matthew Stewart et al.

Opinion
Machine Learning Sensors

A design paradigm for the future of intelligent sensors.

IMAGE BY ANDREY SUSLOV

THE LAST DECADE has seen a surge in commercial applications using machine learning (ML). Similarly, marked improvements in latency and bandwidth of wireless communication have led to the rapid adoption of cloud-connected devices, which gained the moniker Internet of Things (IoT). With such technology, it became possible to add intelligence to sensor systems and devices, enabling new technologies such as Amazon Echo, Google Nest, and other so-called "smart devices." However, these devices offer only the illusion of intelligence and are merely vessels for submitting and receiving queries from a centralized cloud infrastructure. This cloud processing leads to concerns about where user data is being stored, what other services it might be used for, and who has access to it.[7]

More recently, efforts have progressed in dovetailing the domains of IoT and machine learning to embed intelligence directly on the device, known as tiny machine learning (TinyML).[10] TinyML has several benefits over traditional cloud-based IoT architectures as the performance of these devices is both latency- and bandwidth-dependent. For example, wireless communication is associated with high power consumption due to the electric current required to amplify an antenna's signal. Furthermore, potentially sensitive data is being broadcast over large distances, opening up the opportunity for interception by malicious actors. In contrast, TinyML can process data on-device, meaning wireless communication is unnecessary. Such offline devices can improve security, reduce power consumption, and reserve communication solely for firmware updates or communicating anomalies.[1,6] However, this new ML paradigm is met with similar challenges to the IoT workflow, most notably data privacy and the need for more transparency. For more on TinyML, see Prakash et al. on p. 68.

These challenges are best illustrated through example; consider an everyday use case of TinyML known as person detection—determining whether a person is present within an image. One could imagine having "person detectors" spread about their home to control lighting or other domestic systems. The key data-generating device for a person detector is a camera, the data from which is then submitted to the application processor and run through a neural network stored in flash memory. The output from the network then provides a simple binary label denoting whether a person is present.

With current cloud-based IoT deployments, while the binary output of such a device likely does not contain large amounts of sensitive information, there is no way to guarantee that the raw images collected by these devices are not being harvested and used for other purposes, perhaps unsavory. One could imagine that such images might be used to determine how many people live

in a house, their ethnicities, genders, habits, and so forth, all based on data obtained under the guise of person detection. Such factors have evoked concern from privacy advocates, end users, and workers' unions. Unfortunately, this is not an isolated issue, with many newer devices containing microphones or cameras without the user's knowledge.[2] This trend underscores the need for improved transparency and the introduction of regulations or mechanisms to protect user privacy and inform users of what data their devices are collecting.

To this end, we propose the ML sensor, a logical framework for developing ML-enabled embedded systems that empowers end users through its privacy-by-design approach. By limiting the data interface, the ML sensor paradigm helps ensure no user information can be extracted beyond the scope of the sensor's functionality. Our proposed definition is: "An ML sensor is a self-contained, embedded system that utilizes machine learning to process sensor data on-device—logically decoupling data computation from the main application processor and limiting the data access of the wider system to high-level ML model outputs."

The underlying idea behind a ML sensor is that an ML processor is coupled to a sensor as part of a single physical entity, separate from the central processor (see Figure 1). Similar to the distinction between user and kernel space in computer science, these two processors would live in fundamentally different worlds. Instead of the central processor having access to all the data, it would only receive the output from the ML sensor. On the other hand, the ML sensor has access to the data but only contains the functionality and peripherals to perform its essential operation, no other auxiliary computations. *This way, the raw sensor data would never leave the ML sensor, promoting privacy while enabling intelligence.*

This new ML paradigm is met with similar challenges to the IoT workflow, most notably data privacy and the need for more transparency.

For our motivating person-detection example, this could be implemented by passing a binary label denoting whether a person is present in an image. This approach would enable a simple hardware design containing only three communication pins: one for ground, one for power, and one for the binary output of the neural network. Sensors that require more than a single bit for output variables, such as for multiclass classification, would require more channels but only enough to cover the co-domain of the machine learning output. Thus, while the ML sensor is best suited for applications with a finite set of model outputs, this still admits a wide variety of possible commercial applications, from voice command interfaces to analog display monitoring.[9] Useful Sensors recently developed a person detection ML sensor module[7] that operates similarly to our motivating example.

To further promote privacy, the ML learning model could be loaded onto the ML sensor during production without hardware capabilities to enable model updates. Alternatively, secure networking capabilities could be included to ensure the device's firmware and model would still be updateable but would be performed over-the-air and under rare circumstances. Furthermore, the ML Sensor could also incorporate existing privacy-preserving technologies (for example, SGX and TrustZone[5]). However, most current devices within the scope of tinyML are too resource-constrained to implement such technologies.

The ML sensor also has additional desirable features. The device would be self-contained and modular, allowing it to interface easily with embedded systems. This approach makes the technology more accessible to non-experts by removing the need for hardware, software, and ML expertise to develop ML-enabled embedded devices. These devices could also be integrated into larger, more complex systems to enable advanced functionality while providing users assurance that their sensitive data is protected.

An ML sensor would need to be provided alongside a datasheet that communicates its coree sign to ensure ease of use. This datasheet should include sections seen in traditional sensor datasheets (for example, electrical characteristics), as well as additional sections to outline ML model performance, end-to-end performance analysis, and articulate features regarding privacy, ethical, and environmental considerations, including the dataset used for training, inspired by Gebru et al.[3] Such a datasheet could be audited to ensure it meets specific requirements regarding algorithmic bias, device reliability, and performance verification. It would not only provide relevant information to device designers but also help to promote trust

Figure 1. The ML sensor paradigm. The ML model is tightly coupled with the physical sensor, separate from the application processor in the ML sensor. This paradigm provides isolation of data-level computation from the wider system, precluding system-level accessibility to sensitive user data.

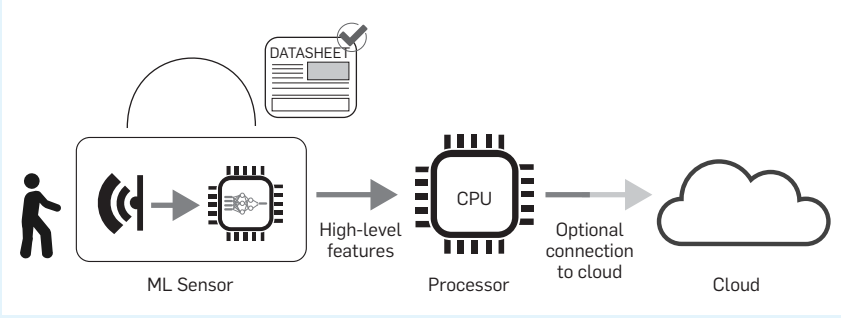


Figure 2. An illustrative example of an ML sensor datasheet.

On the top are the items found in standard datasheets: the description, features, use cases, diagrams and form factor, hardware characteristics, communication specification, and pinout. The bottom includes the new items that must be included in an ML sensor datasheet: the ML model characteristics, dataset nutrition label, environmental impact analysis, and end-to-end performance analysis. While this datasheet is compressed into one page, in a veritable datasheet, sections might be substantially longer and differ significantly based on the device specification and real-world application.

PA1 Person Detection Module

Description: The PA1 Person Detection Module enables you to quickly and easily add smarts to your IoT deployment to monitor and detect for humans. You can use this module indoors and outdoors to understand where and when humans arrive at your deployment site.

Features:

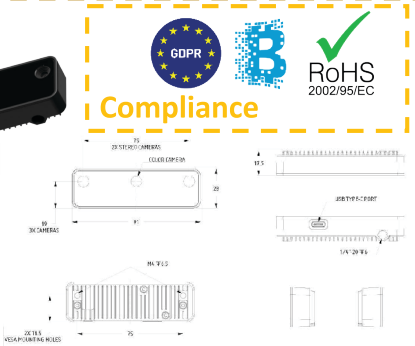
- Real-time Person Detection with On-Device ML
- Indoor and Outdoor use
- Finds a person at a maximum distance of 10 meters to a minimum distance of 5 centimeters
- Operates in low and high light environment (1–2000 Lux) across a wide temperature range (0 to 50°C)
- Features Color and Black-and-White Detection Modules

Use Cases:

- Smart business and home security systems
- Multi-modal key word spotting for virtual assistants
- Occupancy sensors and other infrastructure sensors

Description, Features, and Use Cases

Source: docs.luxonis.com



Diagrams and Form Factor

Compliance

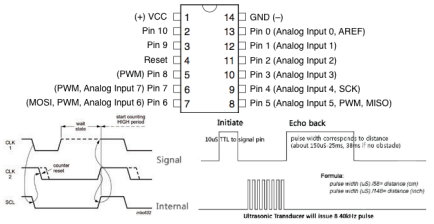
Camera Specs	Color camera	Stereo pair
Sensor	IMX214	OV7251
DFOV / HFOV / VFOV	81° / 69° / 54°	86° / 73° / 58°
Resolution	13MP (4208x3120)	480P (640x480)
Focus	AF: 8cm - ∞ OR FF: 50cm - ∞	Fixed-Focus 6.5cm - ∞
Max Framerate	60 FPS	200 FPS
F-number	2.2 ± 5%	2.2
Lens size	1/3.1 inch	1/7.5 inch
Effective Focal Length	3.37mm	1.3mm
Distortion	< 1%	< 1.5%
Pixel size	1.12µm x 1.12µm	3µm x 3µm

SYMBOL	RATING	MIN	MAX	UNIT
V_{BUS}	Recommended Input Supply Voltage	4.75	5.25	V
V_{BUS_MAX}	Maximum Input Supply Voltage	3.5	5.5	V
I_{VBUS_MAX}	Maximum Input Supply Current	0	1.5	A
P	Power Required	4	6	W
P_{IDLE}	Idle Power Draw	2.4	2.6	W
T_A	Ambient Operating Temperature	45	55	°C

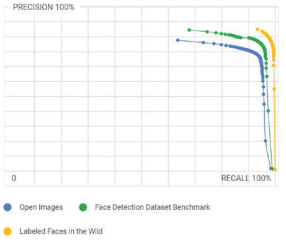
Hardware Characteristics

Source: docs.luxonis.com

Sources: fabacademy.org, electroschematics.com, and nxp.com/docs



Communication Specification and Pinout



Model performance: Measured with Precision-Recall (PR) and Area Under the PR Curve (PR-AUC). Download raw performance results data here. Disaggregated performance measured with Recall, which captures how often the model misses faces with specific characteristics. Equal recall across subgroups corresponds to the "Equality of Opportunity" fairness criterion.

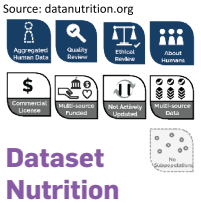
Performance evaluated on:

- A subset of Open Images
- Face Detection Dataset and Benchmark
- Labeled Faces in the Wild

Source: modelcards.withgoogle.com

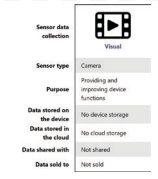
Model Characteristics

Source: datanutrition.org



Dataset Nutrition Label

Source: iotsecurityprivacy.org



IoT Security and Privacy Label

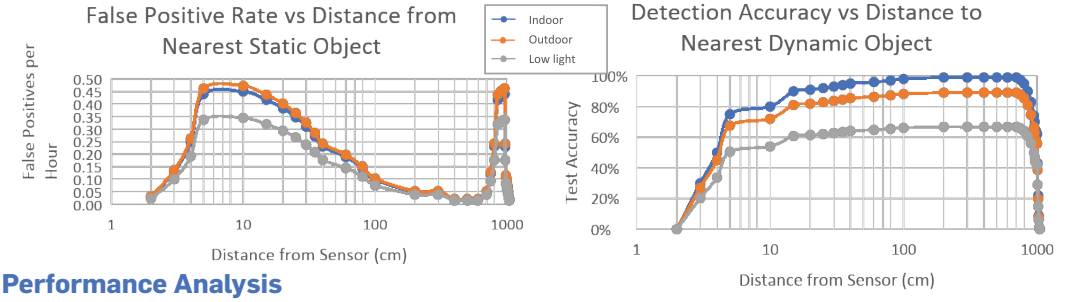
Environmental Impact: Full report can be found here.

390g CO_2-eq

23l Water

Source: st.com

Environmental Impact



Performance Analysis

between corporations and end users based on added transparency and validation by a third-party entity.

Figure 2 is an example of what such an ML sensor datasheet might look like for our example person detector module. Designing such datasheets requires community involvement and engagement since it affects stakeholders at all levels, including end users, manufacturers, technical experts, privacy advocates, auditing bodies, and policymakers. The ML sensor ecosystem will need to decide:

- What high-level information needs to be communicated to the end user when purchasing a device utilizing ML sensors, such as what sensors a device contains and what the data it collects is used for (that is, data nutrition label[4]).
- What ethical or compliance standards (for example, GDPR, RoHS) must be met by manufacturers and corporations utilizing ML sensors.
- What categories and information must be communicated in a datasheet, and how it should be communicated.

We believe the net impact afforded by increasing the usability of ML in hardware applications and its possible positive downstream effects on privacy, security, and transparency will be positive. However, as with any approach, this paradigm has challenges, and the net positive impact relies on developers applying appropriate ethical considerations when designing and developing ML sensors. For example, traditional ML concerns remain, such as model bias, the potential for adversarial attacks, and reduced explainability of the device's functionality. There is also the potential for ML sensors to be exploited for malicious purposes, such as within weaponry or suppressing freedom of speech. Based on these shortcomings, we envisage these devices being used to augment existing systems and not to replace them, especially for safety-critical applications. And perhaps someday, it might be possible to walk into a hardware store and purchase a person detection sensor like one might purchase a temperature sensor today.

A Call to Action

The rapid development of intelligent sensors means the challenges described here will occur faster than most anticipate. A privacy-by-design approach is necessary to tackle these issues proactively and promptly. Simultaneously, a dialogue is needed between the general public, corporations, manufacturers, and policymakers, to discern the appropriate level of privacy, security, and transparency to meet the needs of all parties and how these needs might be achieved. The solution may involve developing an auditing system, certification process, regulatory body, or a mixture of these alongside other mechanisms. Care must be taken to ensure user privacy and security are protected while avoiding stifling innovation. There may be lessons that can be learned from similar ethical issues within cloud-based ML, as well as solutions made for embedded ML that may transfer over to the domain of cloud-based ML. C

References

1. Banbury, C.R. et al. (2021, January 29). Benchmarking tinyml systems: Challenges and direction. arXiv.org. https://arxiv.org/abs/2003.04821
2. Fussell, S. The microphones that may be hidden in your home. *The Atlantic* (Feb. 23, 2019); https://bit.ly/3rlnOoR
3. Gebru, T. et al. Datasheets for datasets. *Commun. ACM 64*, 12 (Dec. 2021); https://doi.org/10.1145/3458723
4. Holland, S. et al. The dataset nutrition label. *Data Protection and Privacy.* (2020); https://doi.org/10.5040/9781509932771.ch-001
5. Kuznetsov, E. et al. Securefl: Privacy preserving federated learning with sgx and trustzone. In *Proceedings of 2021 IEEE/ACM Symposium on Edge Computing (SEC)* (Dec. 2021), 55–67.
6. Reddi, V.J. et al. Widening access to Applied Machine Learning with tinyml. *Harvard Data Science Review* (2020); https://doi.org/10.1162/99608f92.762d171a
7. Tawalbeh, L. et al. IoT privacy and security: Challenges and solutions. *Applied Sciences, 10*, 12 (2020); https://doi.org/10.3390/app10124102
8. Useful Sensors Inc. Person Sensor (Aug. 1, 2023); https://bit.ly/3t7fBoL
9. Warden, P. et al. Machine Learning Sensors. (2020); arXiv preprint arXiv:2206.03266.
10. Warden, P. and Situnayake, D. *TinyML: Machine Learning with Tensorflow Lite on Arduino and Ultra-Low-Power Microcontrollers.* O'Reilly. (2020).

Pete Warden (pete@petewarden.com) is a Ph.D. student at Stanford University, Stanford, CA, USA, and CEO of Useful Sensors, Mountain View, CA, USA.

Matthew Stewart (matthew_stewart@g.harvard.edu) is a postdoctoral researcher at Harvard University, Cambridge, MA, USA.

Brian Plancher (bplancher@barnard.edu) is an assistant professor of computer science at Barnard College, Columbia University, New York, NY, USA.

Sachin Katti (skatti@stanford.edu) is an associate professor of computer science and electrical engineering at Stanford University, Stanford, CA, USA.

Vijay Janapa Reddi (vj@eecs.harvard.edu) is an associate professor of computer science and electrical engineering at Harvard University, Cambridge, MA, USA.