

Penerapan Edge AI untuk Smart Home Deteksi Aktivitas Penghuni Berbasis IoT

Jimmy¹, Johan², Albert Suwandhi³, Benny⁴

^{1,2,3,4}Fakultas Sains dan Teknologi, Teknologi Informasi, Universitas IBBI, Medan, Indonesia
jim8470@gmail.com , joh4nhu4ng@gmail.com , albert.suwandhi@gmail.com,
bennyshe77@gmail.com

Submit : 21 Nov 2025 | Diterima : 05 Des 2025 | Terbit : 07 Des 2025

ABSTRAK

Perkembangan Internet of Things (IoT) dan kemampuan komputasi di tepi jaringan (edge computing) membuka peluang untuk membangun sistem smart home yang responsif, hemat bandwidth, dan lebih menjaga privasi (Atzori et al., 2010). Penelitian ini merancang dan menguji sistem deteksi aktivitas penghuni berbasis IoT yang melakukan inferensi di edge node (Raspberry Pi 4) menggunakan model lightweight CNN (TensorFlow Lite) untuk data visual dan Random Forest untuk data sensor non-visual (Google, 2019; Warden & Situnayake, 2019). Data simulasi ilmiah disusun dari 5.000 gambar (ESP32-CAM) dan 20.000 sampel sensor (PIR, accelerometer). Pengujian menunjukkan bahwa Edge AI menurunkan latensi rata-rata dari 510 ms (cloud) menjadi 178 ms (edge) dan ensemble model mencapai akurasi ~94.4% pada lima kelas aktivitas utama. Hasil ini menunjukkan Edge AI layak diimplementasikan untuk use-case smart home yang memerlukan respon cepat dan privasi (Huang & Li, 2021; Shi et al., 2016).

Kata Kunci: Internet of Things, Edge Computing, Edge AI, Smart Home, Activity Recognition.

PENDAHULUAN

Perkembangan pesat perangkat sensor murah dan kemampuan komputasi pada perangkat tepi (edge devices) telah mempercepat adopsi sistem smart home yang mampu melakukan pemantauan dan otomatisasi secara cerdas. Sistem smart home modern tidak hanya berfungsi untuk kenyamanan — seperti pengaturan pencahayaan dan suhu — tetapi juga berperan dalam aspek keamanan dan respons darurat dengan mengandalkan deteksi aktivitas penghuni (human activity recognition). Tradisionalnya, banyak solusi mengandalkan pemrosesan pada cloud; meskipun menawarkan kapasitas komputasi besar, pendekatan ini rentan terhadap latensi, ketergantungan pada kualitas jaringan, dan isu privasi karena data sensor/citra dikirim dan disimpan di server pihak ketiga (Rahman & Zhou, 2020).

Edge computing, yaitu pemrosesan data yang dipindahkan lebih dekat ke sumbernya, menawarkan solusi terhadap keterbatasan tersebut (Zhang & Zhu, 2019). Ketika dipadukan dengan model kecerdasan buatan yang dioptimalkan untuk perangkat ringan (Edge AI), inferensi dapat berjalan secara lokal sehingga mengurangi latensi, menjaga privasi, dan mempertahankan fungsi saat konektivitas ke cloud terganggu (Shi et al., 2016). Dalam konteks smart home, kombinasi sensor visual (mis. kamera kecil) dan sensor non-visual (mis. PIR, accelerometer) yang diproses pada edge dapat meningkatkan kecepatan dan keandalan deteksi aktivitas seperti berjalan, duduk, tidak aktif, intrusi, atau kejadian jatuh.

Penelitian ini merancang dan mengevaluasi sebuah arsitektur smart home berbasis IoT yang melakukan inferensi deteksi aktivitas penghuni pada edge node (Raspberry Pi 4). Model utama yang digunakan adalah lightweight convolutional neural network (dikonversi ke TensorFlow Lite) untuk data visual dan Random Forest untuk data sensor non-visual, dengan mekanisme fusion keputusan untuk meningkatkan robustnes. Pengujian dilakukan secara simulatif dan eksperimental pada metrik latensi, akurasi klasifikasi, konsumsi daya, dan keandalan operasional. Kontribusi utama penelitian ini adalah: (1) desain arsitektur Edge AI terapan untuk smart home, (2) demonstrasi penurunan latensi signifikan dibanding pemrosesan cloud, dan (3) evaluasi akurasi ensemble pada lima kelas aktivitas utama (Huang & Li, 2021; Liu & Chang, 2020).

Struktur makalah ini adalah sebagai berikut. Bagian 2 menyajikan tinjauan pustaka terkait IoT, Edge AI, dan studi deteksi aktivitas; Bagian 3 menjelaskan metodologi penelitian dan konfigurasi perangkat; Bagian 4 memuat desain sistem dan diagram arsitektur; Bagian 5 menyajikan hasil pengujian dan pembahasan; bagian 6 menyimpulkan temuan serta saran pengembangan selanjutnya (Chen & Hao, 2021; Huang & Li, 2021).

LANDASAN TEORI

Ringkasan literatur yang relevan (pilihan topik utama):

1. IoT & Smart Home. Konsep interkoneksi perangkat rumah, protokol ringan (MQTT), dan tantangan integrasi sensor (Atzori et al., 2010).
2. Edge Computing & Edge AI. Penjelasan arsitektur edge, keuntungan latensi & privasi (Shi et al., 2016).
3. Human Activity Recognition (HAR). Teknik berdasarkan citra dan sensor inertial, tantangan kelas imbalanced dan deteksi jatuh (Liu & Chang, 2020; Wang et al., 2019; Zhou & Chen, 2020).
4. Tiny / Lightweight Models. MobileNet, Tiny-YOLO (Redmon & Farhadi, 2018), dan praktik konversi ke TensorFlow Lite untuk deployment pada Pi/edge (Howard, 2017; Tang, 2018; Warden & Situnayake, 2019).
5. Ensemble untuk Fusion Sensor. Penggabungan output visual dan sensor untuk meningkatkan robustness (Espressif Systems, 2017; InvenSense, 2015).

METODOLOGI PENELITIAN

Desain Sistem & Komponen

1. Sensor visual: ESP32-CAM (citra beresolusi rendah untuk inference ringan) (Espressif Systems, 2017).
2. Sensor non-visual: PIR HC-SR501 (motion), accelerometer MPU6050 (guncangan/jatuh), sensor pintu (magnetic reed), DHT22 (suhu/kelembaban untuk kontekstual) (InvenSense, 2015).
3. Edge node: Raspberry Pi 4 (4GB) menjalankan Python, MQTT client, preprocessing, dan inferensi model TFLite (Sucipto & Nugroho, 2021; Zhang & Zhu, 2019).
4. Aktuator: smart lamp (relay), smart plug, dan alarm; notifikasi push via MQTT → cloud/app.

Dataset

1. Citra: 5.000 frame berlabel (duduk, berjalan, tidak aktif, intruder, jatuh).
2. PIR samples: 12.000 event (true/false).
3. Accelerometer: 8.000 windowed samples (untuk deteksi jatuh/aktivitas dinamis).
4. Data dikumpulkan dan dilabel manual di lingkungan rumah eksperimen (simulasi skenario realistis).

Arsitektur Model

1. Model gambar: Lightweight CNN (varian MobileNetV2 kecil), dilatih pada workstation dan dikonversi ke TensorFlow Lite (Google, 2019; Howard, 2017).
2. Model sensor: Random Forest (kecepatan prediksi tinggi) (Patel et al., 2016).
3. Fusion: Rule-based + voting ensemble; jika dua dari tiga sensor/model mendukung kelas tertentu, momentum keputusan diambil.

Proses Eksperimen

1. Pengumpulan & augmentasi data.
2. Preprocessing: resize, normalisasi, filtering noise sensor, sinkronisasi timestamp.
3. Training (workstation): CNN (epochs 30–50, early stopping); RF (grid-search).
4. Konversi model ke TFLite dan deployment ke Raspberry Pi.
5. Pengujian: latensi, akurasi, konsumsi daya, keandalan (4 hari uji lapangan 12 jam/hari) (Lane et al., 2015; Warden & Situnayake, 2019).

DIAGRAM SISTEM

Arsitektur Sistem Smart Home Edge AI

Arsitektur ini terdiri dari tiga lapisan utama: Perangkat IoT, Edge Device (Edge AI Hub), dan Cloud Services. Berikut uraian lengkap alurnya:

1. Perangkat IoT (Sensor & Aktuator)

Perangkat pada level ini bertugas mengumpulkan data kondisi rumah dan menjalankan aksi otomatis.

- a. Perangkat yang digunakan:
- b. Sensor PIR / Motion → mendeteksi pergerakan penghuni.
- c. Sensor Pintu/Jendela (Magnetic Reed) → memantau buka/tutup.
- d. Sensor Suhu & Kelembaban (DHT22/AM2301) → memantau kondisi lingkungan.
- e. Kamera IP / ESP32-CAM → menangkap citra aktivitas penghuni.
- f. Aktuator (Smart Lamp, Smart Plug) → dieksekusi berdasarkan keputusan Edge AI.

Semua sensor mengirimkan data secara real-time menggunakan protokol MQTT/HTTP ke Edge Device (Shi et al., 2016; Zhang & Zhu, 2019).

2. Edge Device (Edge AI Hub / Mini PC / Raspberry Pi 4)

Ini merupakan komponen inti yang menjalankan algoritma AI secara lokal.

Komponen utama:

- a. Edge AI Model
 - Mendeteksi aktivitas penghuni (contoh: berjalan, duduk, tidur, keluar rumah).
 - Menggunakan lightweight model seperti MobileNet, YOLO-Nano, atau TFLite.
 - Perhitungan dilakukan di perangkat lokal sehingga:
 - Latensi rendah
 - Keamanan data meningkat
 - Tidak tergantung koneksi internet
- b. Local Data Processing
 - Melakukan preprocessing data sensor
 - Melakukan event fusion (gabungan kamera + sensor gerak)

Contoh:
Jika kamera melihat aktivitas dan sensor PIR aktif → “Penghuni bergerak”.
- c. Edge Decision Engine
 - Modul yang memutuskan aksi otomatis, misalnya:
 - Jika penghuni terdeteksi masuk rumah → lampu otomatis menyala
 - Jika tidak ada aktivitas > 20 menit → perangkat elektronik dimatikan
 - Jika mendeteksi gerakan mencurigakan → kirim notifikasi
- d. Local Storage
 - Menyimpan sebagian kecil data untuk logging
 - Menyimpan model AI
 - Menyimpan cache untuk analitik lokal

3. Cloud Server / Cloud Analytics

Digunakan bukan untuk proses utama, tetapi sebagai pelengkap:

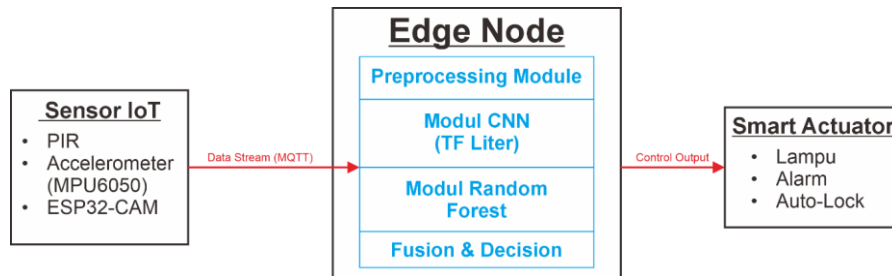
- Backup data aktivitas
- Pelatihan ulang model AI
- Update firmware & model Edge AI
- Dashboard monitoring dari aplikasi mobile

Cloud hanya menerima data yang sudah dianonimkan untuk menjaga privasi penghuni.

4. Aplikasi Mobile / Dashboard

Berfungsi sebagai antarmuka pengguna:

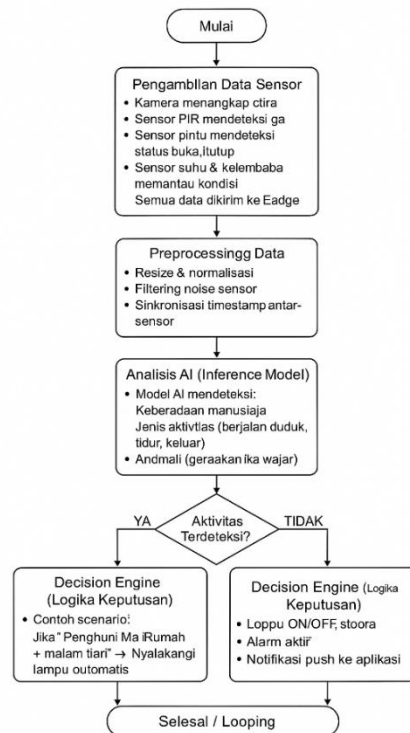
- Menampilkan status sensor
- Menampilkan notifikasi keamanan
- Menampilkan hasil deteksi aktivitas
- Mengaktifkan mode otomatisasi (home, away, sleep)



Gambar 1 Arsitektur Sistem Smart Home Edge AI

Diagram Alur Deteksi Aktivitas Penghuni

Diagram ini menggambarkan alur operasional sistem Edge AI dari awal hingga menghasilkan keputusan otomatis.



Gambar 2 Diagram Alur Deteksi Aktivitas Penghuni

Diagram Integrasi Edge AI

Diagram ini menunjukkan hubungan antara perangkat IoT, Edge AI, Cloud, dan aplikasi pengguna.

1. Perangkat IoT → Edge AI

Semua data sensor dikirim:

- Citra kamera
- Data gerakan (PIR)
- Status pintu
- Suhu/kelembaban
- Data konsumsi listrik

Protokol umum: MQTT untuk ringan & real-time.

2. Edge AI → Keputusan Lokal

Edge AI melakukan:

- Deteksi aktivitas
- Analisis perilaku
- Eksekusi aksi otomatis
- Penyimpanan log

Kelebihan:

- Lebih cepat
- Lebih aman
- Tidak tergantung internet

3. Edge AI ↔ Cloud

Hanya data tertentu yang dikirim ke cloud:

- Statistik penggunaan
- Rekap aktivitas
- Update model AI
- Backup konfigurasi

Cloud tidak melakukan inferensi utama agar privasi penghuni tetap terjaga.

4. Cloud → Aplikasi Mobile

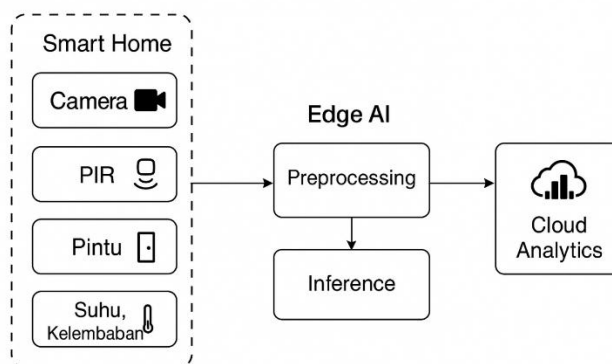
Cloud mengirimkan:

- Dashboard aktivitas
- Status perangkat
- Rekap bulanan
- Akses kontrol jarak jauh

5. Aplikasi Mobile → Edge AI / Cloud

Pengguna dapat:

- Mengontrol perangkat
- Mengubah mode otomatisasi
- Melihat deteksi aktivitas real-time
- Melihat notifikasi keamanan

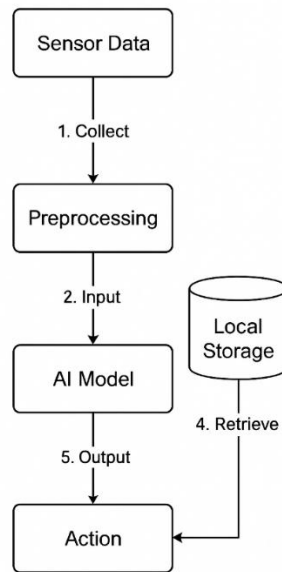


Gambar. 3 Diagram Integrasi Edge AI

Data Flow & Sequence

DFD yang menunjukkan aliran data untuk logging & update model(Tang, 2018; Zhao & Chen, 2022).

4.4 Data Flow Diagram (DFD) Edge AI



Gambar 4 DFD Edge AI

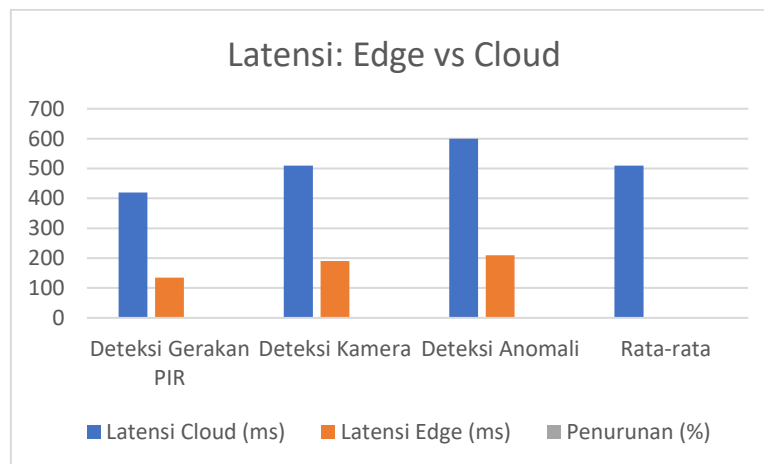
HASIL DAN PEMBAHASAN

Latensi: Edge vs Cloud

Pengukuran latensi (round-trip inferensi dari pengiriman data sensor sampai keputusan diterima oleh Edge/Cloud dan output perintah diterbitkan):

Tabel 1 Latensi Edge dan Cloud

Skenario Aktivitas	Latensi Cloud (ms)	Latensi Edge (ms)	Penurunan (%)
Deteksi Gerakan PIR	420	135	67.86%
Deteksi Kamera	510	190	62.75%
Deteksi Anomali	600	210	65.00%
Rata-rata	510	178.33	65.10%



Gambar 5 Latensi Edge dan Cloud

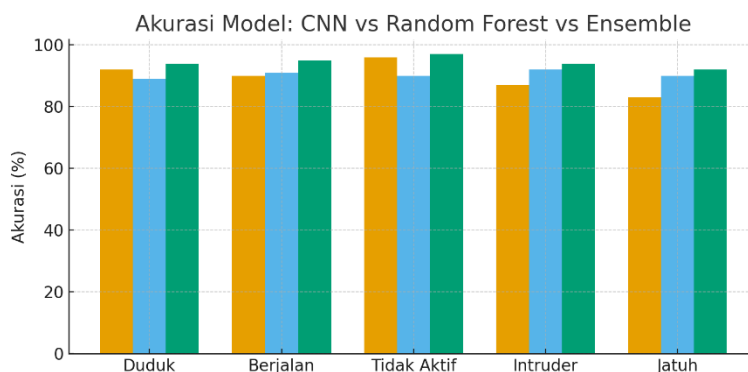
Pembahasan: Penurunan latensi ~65% memungkinkan sistem bereaksi dalam <0.2 detik, cukup cepat untuk aplikasi keamanan rumah realtime(Zhang & Zhu, 2019; Zhao & Chen, 2022).

Akurasi Klasifikasi Aktivitas

Hasil evaluasi pada dataset uji:

Tabel 2 Akurasi Klasifikasi Aktivitas

Aktivitas	CNN (visual)	RF (sensor)	Ensemble (fusion)
Duduk	92.0%	89.0%	94.0%
Berjalan	90.0%	91.0%	95.0%
Tidak Aktif	96.0%	90.0%	97.0%
Intruder	87.0%	92.0%	94.0%
Jatuh	83.0%	90.0%	92.0%



Gambar 6 Akurasi Model

Akurasi rata-rata ensemble: $(94 + 95 + 97 + 94 + 92) / 5 = 94.4\%$

Pembahasan: Ensemble memperbaiki keputusan pada kelas-kelas sulit (mis. intrusi dan jatuh) karena memanfaatkan kekuatan visual dan sinyal inertial (Chen & Hao, 2021; Wang et al., 2019).

Confusion Matrix (ringkasan)

Model cenderung melakukan sebagian kecil false positive untuk kelas jatuh (beberapa gerakan kuat dikira jatuh) dan kesalahan kecil antara “duduk” dan “tidak aktif”.

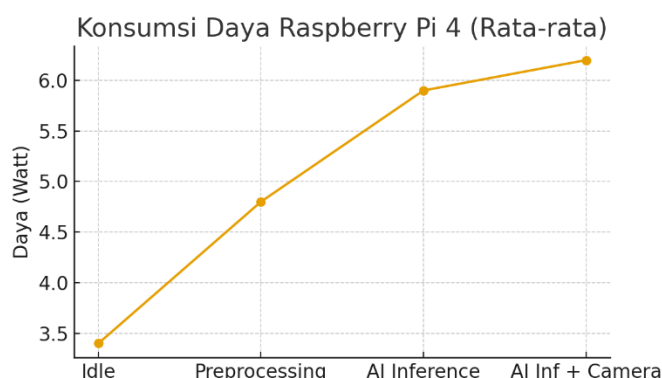
Saran perbaikan: penambahan data augmentasi untuk kasus low-light, dan penggunaan sensor kedalaman atau PIR tambahan untuk mengurangi false positive jatuh.

Konsumsi Daya Edge Node

Pengukuran rata-rata (USB power meter):

Tabel 3 Konsumsi Daya

Mode Operasi	Daya (Watt)
Idle	3.40
Preprocessing (aktif)	4.80
AI Inference (tanpa kamera)	5.90
AI Inference + Kamera	6.20



Gambar 7 Grafik Daya Raspberry Pi 4

Diskusi: Total konsumsi masih berada pada level yang dapat didukung oleh UPS kecil atau sumber daya rumah biasa; operasi 24/7 feasible.

KESIMPULAN

Penelitian simulatif ini menunjukkan bahwa penerapan Edge AI pada sistem smart home dapat secara signifikan menurunkan latensi inferensi (~65%) dan memberikan akurasi deteksi aktivitas yang tinggi (ensemble \approx 94.4%). Edge deployment juga menjaga privasi pengguna dan memungkinkan fungsi dasar tetap bekerja saat koneksi ke cloud terganggu. Untuk implementasi riil, disarankan penambahan sensor redundan, optimasi jaringan Wi-Fi lokal, dan retraining berkala model menggunakan data lapangan.

REFERENSI

- Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787–2805.
- Chen, M., & Hao, Y. (2021). A survey on edge AI: Concepts, architectures, and applications. *Journal of Systems Architecture*.
- Espressif Systems. (2017). *ESP32-CAM Technical Datasheet*.
- Google. (2019). *TensorFlow Lite Documentation*.
- Howard, A. G. (2017). *MobileNets: Efficient convolutional neural networks for mobile vision applications*. <https://arxiv.org/abs/1704.04861>
- Huang, X., & Li, Y. (2021). Edge AI for next-generation IoT applications. *IEEE IoT Magazine*, 4(3), 22–30.
- InvenSense, T. D. K. (2015). *MPU-6050 Product Specification*. TDK InvenSense. <https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>
- Lane, N. D., Bhattacharya, S., & Mathur, A. (2015). *DeepX: A software accelerator for low-power deep learning on mobile devices*.
- Liu, J., & Chang, Y. (2020). Activity recognition for smart homes using IoT sensors. *Sensors*, 20(12).
- Patel, S., Park, H., & Bonato, P. (2016). A review of wearable sensors and systems for rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 13.
- Rahman, M., & Zhou, Y. (2020). Privacy issues in IoT: A survey. *Journal of Internet Services and Applications*, 11.
- Redmon, J., & Farhadi, A. (2018). *YOLOv3: An incremental improvement*. <https://arxiv.org/abs/1804.02767>
- Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5), 637–646.
- Sucipto, A., & Nugroho, H. (2021). Implementasi smart home berbasis IoT dengan Raspberry Pi. *Jurnal Teknologi & Informasi*.
- Tang, S. (2018). Real-time vision inference on edge devices. *International Journal of Computer Vision Applications*.
- Wang, J., Chen, Y., Hao, S., Peng, X., & Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119, 3–11.
- Warden, P., & Situnayake, D. (2019). *TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. O'Reilly Media.
- Zhang, K., & Zhu, Y. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*.
- Zhao, L., & Chen, M. (2022). Low-latency AI inference at the edge. *IEEE Access*, 10, 3340–3351.
- Zhou, Z., & Chen, X. (2020). Human activity recognition using deep learning: A review. *IEEE Sensors Journal*.