

Parallel Feature Fusion for Multi-Modal Scene Recognition on Dual-Core MCUs

Kieran Woodward, *Department of Computer Science, Nottingham Trent University, Nottingham, UK*

Eiman Kanjo, *Computing Department, Imperial College London, London, UK*

Abstract—Multi-modal sensing promises more robust environmental understanding for pervasive computing applications, but implementing sophisticated sensor fusion on resource-constrained devices remains challenging. We present a novel approach that leverages dual-core microcontrollers to enable parallel processing of visual and audio data for scene recognition. Our system concurrently executes specialized neural networks across separate cores while efficiently fusing their intermediate features, achieving 93% classification accuracy across nine different environments. This represents a 12.27% improvement over single-modality approaches while reducing latency by 48ms compared to sequential processing. The entire system operates within just 258KB of memory, demonstrating that complex multi-modal AI is achievable even on highly constrained devices. By enabling sophisticated environmental understanding without cloud connectivity, this work advances privacy-preserving edge intelligence for a wide range of future applications.

Edge computing opens up the potential for small, battery-powered device that can simultaneously see and hear its environment, understanding complex scenes just like humans do. While cloud-based AI systems have long been able to process multiple types of sensory data, bringing this capability to tiny, power-constrained devices opens up exciting new possibilities for pervasive computing. Recent advances in embedded artificial intelligence have enabled running deep learning models directly on microcontrollers and other resource-constrained edge devices [7]. However, complex multi-modal, multi-model classification tasks still pose challenges due to the limited memory, compute, and power available outside of the cloud.

Multi-modal machine learning combines multiple data types and sensor inputs for a more comprehensive understanding of complex environments. Most prior multi-modal fusion works transmit raw sensory streams to the cloud for processing or are not suitable for real-world deployment [9, 3]. However, offloading multi-modal analysis incurs significant latency and raises privacy concerns. Edge computing has gained

popularity for performing inference closer to the data source, avoiding the drawbacks of cloud-based machine learning [18]. However, embedded devices at the extreme edge pose challenges for complex neural networks due to substantial memory constraints and restricted power.

TinyML research focuses on model optimization techniques for resource-constrained deployment [1]. More hardware-centric works also propose sensor data encoding methods or explore specialized system-on-chip accelerators [15]. However, while these approaches enable single-model deployment on one core, this can waste hardware resources because the other core sits idle when not in use. The simultaneous execution of multiple specialized models on separate cores, crucial for utilizing available resources for sophisticated multi-modal analysis, remains largely unexplored despite advances in microcontroller capabilities.

While existing work has explored model parallelisation and fusion strategies in cloud and mobile environments, these approaches typically assume abundant memory and shared memory architectures that don't exist on microcontrollers [12]. Our work demonstrates a novel approach to this challenge by leveraging an often-overlooked capability of modern microcontrollers, multiple processing cores. We propose an alternative approach that splits computational tasks across dual

processing cores available even on highly constrained microcontrollers. We developed a technique to concurrently run separate uni-modal models specializing in different classification tasks and fuse the features and outputs across models after parallel on-device execution.

Specifically, we explore early feature-level fusion for efficiently combining on-device scene recognition by partitioning computational loads across available cores. We concurrently run separate models, each operating on distinct modalities (image and audio), across multiple processing units while fusing their second-to-last activations into a joint embedding to improve classification performance and reduce latency. We demonstrate the feasibility of this approach using the Portenta H7's dual-core architecture, which couples an ARM Cortex-M7 processor with an ARM Cortex-M4 processor for independent neural network inferences.

While parallel execution of neural networks is well-established [2], implementing it effectively on dual-core microcontrollers is not a simple task due to severe hardware constraints. Basic operations like storing activation tensors or sharing features between cores can easily exceed available resources, requiring careful balance between parallelisation granularity and communication overhead. Unlike traditional implementations that leverage sophisticated scheduling algorithms, we are limited to simple coordination mechanisms due to restricted compute capabilities.

This work has significant implications for pervasive computing, enabling:

- Local processing of multiple sensory inputs, ensuring privacy and reducing bandwidth costs
- Extended battery life through efficient parallel processing
- Real-time response to complex environmental changes
- Deployment of sophisticated AI capabilities on low-cost hardware

In this article, we'll explore how this dual-core approach works, examine its real-world performance, and discuss its implications for the future. We discuss practical insights and demonstrate how our method achieves a 12.27% increase in classification accuracy compared to single-modality approaches while reducing latency through the parallel processing approach.

The Challenge of Multi-Modal Edge AI

As our world becomes increasingly connected, there's a growing need for devices that can understand their

environment comprehensively. Just as humans naturally combine what we see and hear to understand a scene, modern AI systems aim to fuse multiple types of sensory data for better environmental awareness. However, implementing this multi-modal perception on resource-constrained devices presents several challenges.

The Multi-Modal Imperative

Multi-modal machine learning attempts to mirror human perception by combining diverse data types and sensor inputs for a more enhanced understanding of complex environments. By fusing data from various modalities, such as imagery, audio, and sensors, we can improve inference for complex tasks [13]. For instance, in scene recognition, visual data alone might struggle to distinguish between a quiet beach and a crowded one, while audio data could instantly reveal the difference. Similarly, differentiating between a restaurant and a grocery store becomes more reliable when combining both visual and acoustic information.

There are many scenarios where multi-modal inference may make a vital difference such as healthcare Monitoring where a smart wearable device monitoring an elderly person needs to combine visual, audio, and motion data to accurately distinguish between someone peacefully sleeping and a potential fall or medical emergency. Additionally, multi-model sensing could benefit industry safety as in a factory setting, equipment failure detection becomes significantly more reliable when combining visual inspection with acoustic monitoring. A machine might look normal in camera feeds but emit subtle acoustic signatures indicating imminent failure. Traditional single-sensor systems often miss these early warning signs.

Resource Constraints and Edge Processing

Traditional approaches to multi-modal fusion often rely on cloud processing, transmitting raw sensory streams for analysis [14, 11]. However, this creates several critical issues with privacy, latency and reliability.

Edge computing has emerged as a solution, bringing computation closer to the data source [7]. However, embedded devices at the extreme edge also face severe constraints including limited memory typically less than 2MB of internal flash and 1MB of RAM, limited computational capabilities and strict power consumption requirements for battery operation.

The field of TinyML has made significant progress in enabling AI on constrained devices through various optimization techniques such as quantization to reduce model precision, pruning to remove unnecessary

connections, knowledge distillation for creating smaller efficient models and hardware-specific optimizations for specialized accelerators [5, 6]. Despite these advances, running multiple neural networks on a single microcontroller for different modalities remains challenging. The traditional approach of sequential processing creates high latency, while attempting to combine all modalities in a single model often exceeds memory constraints and sacrifices accuracy.

Recent research has explored sharing network layers to enable multi-modal, multi-model execution directly on devices [8]. However, achieving both high accuracy and efficiency for multi-modal fusion remains challenging due to memory limits and computational constraints outside of servers. Intermediate feature fusion, which combines internal representations rather than just fusing decisions from separate models per modality at the output, has shown promise. Features encoded in hidden layers retain greater information content than raw inputs before output decisions [16]. Feature fusion concatenates features from various modalities before final classification, with recent works exploring fusing embeddings across deeper shared layers with gating and transformation operations [10]. However, implementing such fusion strategies on constrained devices presents additional challenges such as memory overhead to store intermediate features and the communication cost of transferring data between cores.

These challenges have historically forced the choice between cloud processing with its associated drawbacks or simplified edge implementations that sacrifice accuracy and capability. However, modern microcontrollers with multiple processing cores offer new possibilities for addressing these challenges through parallel processing and efficient feature fusion. We propose the concurrent classification of two specialized deep neural networks on the separate processing cores of a microcontroller and explore techniques to fuse their intermediate feature representations before final classification.

System Architecture

We propose the concurrent classification of two specialised deep neural networks on the separate processing cores of a microcontroller and explore techniques to fuse their intermediate feature representations before final classification. Our proposed approach explores dual-model feature fusion techniques that can classify complex multi-modal data in a more compute and memory-efficient manner, helping to reduce processing time. The aim is that by splitting com-

putational tasks across processing cores and fusing partial inferences, we can achieve enhanced overall accuracy without increasing latency or substantially growing model size or computational requirements.

Our methodology involves concurrently running two models, each focusing on analysing one data modality, on separate cores available even in low-powered devices. For example, Modality A could be imagery, while Modality B is audio data. The models ModalityNetA and ModalityNetB have their architectures optimized for their respective modalities. The feature extraction from the visual model (ModalityNetA) on Core 1 can be represented as:

$$\mathbf{f}_A = \phi_A(\mathbf{x}_A; \theta_A) \quad (1)$$

where $\mathbf{f}_A$ is the extracted feature vector from the visual model, ϕ_A represents the visual model architecture, $\mathbf{x}_A$ is the input image, and θ_A are the learned parameters of the visual model.

Similarly, the feature extraction from the audio model (ModalityNetB) on Core 2 can be expressed as:

$$\mathbf{f}_B = \phi_B(\mathbf{x}_B; \theta_B) \quad (2)$$

where $\mathbf{f}_B$ is the extracted feature vector from the audio model, ϕ_B represents the audio model architecture, $\mathbf{x}_B$ is the input audio data, and θ_B are the learned parameters of the audio model.

Rather than only leveraging each model's final output class distributions, we fuse their intermediate hidden representations. These feature representations encode higher-level information about input patterns compared to raw data but are more detailed than lightweight final classifications.

Specifically, we extract and concatenate the second-to-last feature layer outputs from ModalityNetA and ModalityNetB. This joint feature vector concatenates richer descriptions of the multi-modal input data from each separate model specialising in one domain. Our feature fusion approach operates on the intermediate representations rather than final classifications. The fusion process can be decomposed into several key steps:

For each modality, we extract features from the penultimate layer:

$$\mathbf{h}_A = \text{ReLU}(\mathbf{W}_{h_A} \mathbf{f}_A + \mathbf{b}_{h_A}) \quad (3)$$

$$\mathbf{h}_B = \text{ReLU}(\mathbf{W}_{h_B} \mathbf{f}_B + \mathbf{b}_{h_B}) \quad (4)$$

where $\mathbf{h}_A$ and $\mathbf{h}_B$ are the hidden representations, and $\mathbf{W}_{h_A}$, $\mathbf{W}_{h_B}$, $\mathbf{b}_{h_A}$, $\mathbf{b}_{h_B}$ are the learned parameters.

The fusion of extracted features is performed through weighted concatenation:

$$\mathbf{f}_{fused} = \mathbf{W}_A \mathbf{h}_A \oplus \mathbf{W}_B \mathbf{h}_B \quad (5)$$

where $\mathbf{f}_{fused}$ is the fused feature vector, $\mathbf{W}_A$ and $\mathbf{W}_B$ are learnable fusion weights for the visual and audio features, respectively, and $\oplus$ represents the concatenation operation. The fused feature vector gets fed into a compact classification sub-network to produce final unified predictions. This step can be represented as:

$$\mathbf{y} = \psi(\mathbf{f}_{fused}; \theta_C) \quad (6)$$

where $\mathbf{y}$ is the predicted class distribution, ψ represents the classification sub-network architecture, and θ_C are the learned parameters of the classification sub-network.

Deployment Methodology

To evaluate the feasibility of the proposed multi-core parallel processing feature fusion approach, we have developed and deployed models to enable scene recognition from combined image and audio data. Our multi-modal classification approach leverages the dual-core processing architecture of the Portenta H7, comprised of an ARM Cortex-M7 processor and ARM Cortex-M4 processor.

We focused on image and audio classification as representative modalities to demonstrate our parallel processing approach as these modalities represent fundamentally different types of data (spatial vs temporal) that benefit from specialised neural network architectures, making them ideal for showcasing how separate cores can run modality-specific models. While our methodology could be extended to other modalities and tasks, we selected classification specifically because it provides a clear benchmark for evaluating the benefits of multi-modal fusion while remaining feasible to deploy on resource-constrained devices. The proven success of CNN architectures for both image and audio classification also allowed us to focus on the core challenges of parallel processing and feature fusion.

Dataset

To evaluate our proposed multi-modal parallel execution fusion approach, we utilized the dataset introduced by [4]. This dataset was specifically designed for multi-modal scene classification, providing synchronized image and audio data for 9 different environments.

The dataset comprises of 16,000 images (128x128x3) and their corresponding 16,000 seconds of audio data where the Mel-Frequency Cepstral Coefficients (MFCC) features were extracted. The

environments covered in the dataset include Beach, City, Forest, River, Jungle, Football Match, Classroom, Restaurant, and Grocery Store. Each environment has a varying number of sources, ranging from 3 to 8, ensuring diversity within each class.

The dataset presents a challenging scene classification problem due to the similarities between certain environments. For example, the Beach and River classes share similar visual and audio characteristics, while the Restaurant and Grocery Store classes both contain a large number of people and speech audio. This dataset allows for a comprehensive evaluation of multi-modal classification approaches in complex real-world scenarios.

Model Architecture

A two-dimensional convolutional neural network (2D CNN) comprising of two convolutional layers, pooling and two fully-connected layers as shown in Figure 1 has been trained on the image data. This model has been deployed on the more powerful Cortex-M7 core to classify images. On the other hand, on the less powerful Cortex-M4 core we utilize a one-dimensional convolutional neural network (1D CNN) comprising of two convolutional layers, pooling and a final dense layer to classify audio. We instantiate the models within separate threads on their respective cores. A final MultiLayer Perceptron (MLP) classification model is comprised of two dense layers to provide the final classification output from the concatenated vector inputs from the image and audio classification model features.

The models' neural network architectures and weights are optimized and compressed using int8 post-training quantisation to meet the computational requirements for deployment on the Portenta H7 with 2MB of internal flash and 1MB of internal RAM. All of the models were then exported as header files to be deployed onto the microcontroller.

The implementation requires careful consideration of hardware constraints and optimization techniques:

- **Model Quantization:** Both models use int8 quantization to reduce memory footprint
- **Memory Management:** Feature vectors are compressed to 8-bit integers
- **Parallel Processing:** Task scheduling is optimized for minimal latency:

$$t_{total} = \max(t_A, t_B) + t_{fusion} \quad (7)$$

where t_A and t_B are processing times for each modality.

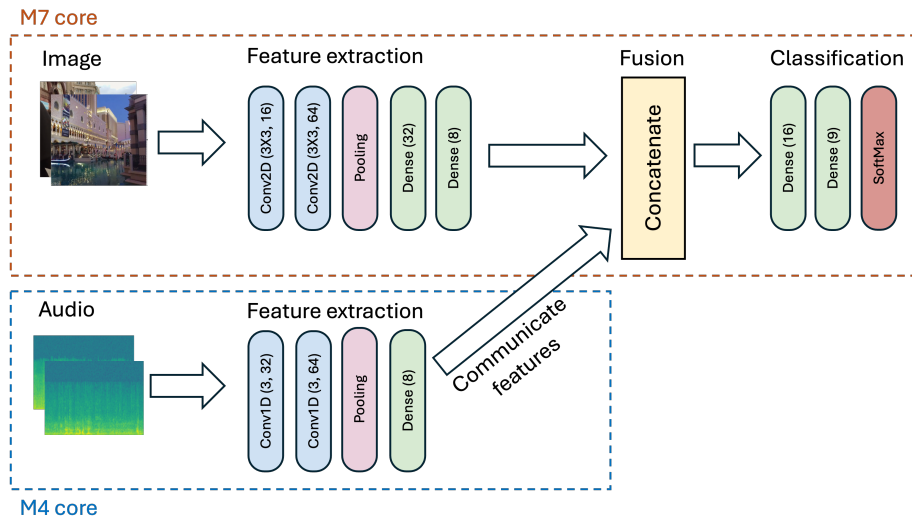


FIGURE 1. System architecture showing the multi-modal dual core feature fusion methodology.

Feature Fusion

The selection of an optimal fusion point is a critical design decision that significantly impacts both performance and resource utilisation. We carefully evaluated several potential fusion strategies before selecting the second-to-last layer as our fusion point. Early fusion strategies, which combine raw or minimally processed inputs, were considered but rejected due to excessive memory requirements for transmitting high-dimensional data between cores and the loss of modality-specific feature learning. Mid-level fusion approaches, combining features from intermediate layers, offered potentially richer feature representations but proved impractical due to the large activation tensors that would need to be shared across cores. Late fusion strategies that only combine final class predictions were also evaluated but showed limited accuracy improvement over single modality models since they fail to capture cross-modal interactions. Our chosen approach of fusing compressed feature embeddings from the second-to-last layer is ideal as it contains high-level semantic features that have been processed through the depth of each network while maintaining reasonable dimensionality (32-64 dimensions). Furthermore, by our addition of a dense embedding layer before fusion, we further reduce the communication overhead between cores to just 8 dimensions per inference while preserving the most discriminative features.

One of the key challenges faced in implementing the proposed dual-core multi-model fusion architecture on a microcontroller was the inability to directly access

intermediate tensors of the output features between the separate processing cores. Many microcontrollers have isolated memory regions for each core without a unified shared address space. This means that each core operates within its own separate memory region, rather than sharing a common memory pool. However, while cores cannot directly access each other's memory, they can communicate data between their separate memory regions through specific interfaces. This memory separation provides hardware protection but limits efficiently accessing features across cores during the parallel execution.

To circumvent this limitation, we devised an approach to extract the higher-level semantic features from each specialized model before transmission to the opposing core for fusion. Specifically, we added an additional dense layer at the end of each model's architecture to generate a compressed feature vector embedding. However, we did not include the typical final softmax layer, as the goal was not to produce classification outputs but rather to encode small feature representations from each modality.

These smaller output feature vectors from the final dense layer can then be transmitted across cores, requiring substantially less memory bandwidth compared to sharing the full intermediate tensors. On the receiving core, these compact feature representations from both models are concatenated together into a joint multi-modal embedding vector as shown in Figure 1.

This fused embedding is then fed into a lightweight shared MLP classification network to produce the final multi-modal predictions. While introducing some addi-

tional compute from the dense layers, this approach avoids directly sharing full activation tensors while still enabling cross-modal fusion of features extracted from each model's specialized inference.

Results and Performance Analysis

The results presented in Table 1 demonstrate the performance of the individual image 2D CNN and audio 1D CNN models, as well as the fused model. The image CNN achieved an accuracy of 80.73%, while the audio CNN attained an accuracy of 82.47% when evaluated using hold-out validation with a 20% test split. Notably, the fused model, which utilized the extracted feature embeddings from the two uni-modal models as input, achieved a significantly improved accuracy of 93% - a significant improvement of 12.27% over the best-performing single modality system. This improvement highlights the effectiveness of multi-modal fusion in enhancing classification performance.

Model	Accuracy	Model Size
Image (2D CNN)	80.73%	126KB
Audio (1D CNN)	82.47%	91KB
MLP Fusion Layers	-	41KB
Complete System	93.00%	258KB

TABLE 1. Performance comparison of individual components and complete system. The Complete System represents the full multi-modal architecture that combines Image CNN, Audio CNN, and MLP Fusion Layers to achieve improved classification accuracy through parallel feature fusion.

In addition to accuracy, this work also considers the importance of model size to enable deployment on resource-constrained microcontrollers. Each model was exported as a header file (.h) to ensure compatibility with the Arduino Portenta H7 microcontroller. The model sizes ranged from 41 KB to 126 KB, with a combined size of 258 KB. This combined size is well within the 2 MB flash memory available on the microcontroller, making it feasible for deployment. Furthermore, the early feature fusion approach adopted helped reduce memory consumption compared with late feature fusion where the final MLP layers would need to be duplicated across both cores to get a classification output from both models before fusion. This demonstrates that adopting an early feature fusion approach reduced the memory consumption by 41KB. This reduction comes from eliminating the need for duplicate classification layers across cores.

Furthermore, the latency of the models was also a key consideration. The latency of the audio model was 48ms while the latency of the image model was 49ms. This is important to consider as if the models

were run sequentially using a single core rather than adopting the dual core parallel processing approach it would have increase latency to 97ms (the sum of individual model latencies), demonstrating the benefits of parallel processing on microcontrollers. This 48ms reduction in latency proves particularly valuable for real-time applications where immediate response is crucial.

The results demonstrate the successful implementation of feature-level fusion on a resource-constrained microcontroller. The fused model achieves higher accuracy compared to the individual uni-modal models while maintaining a compact model size and reducing latency through parallel processing.

Discussion

Combining data from multiple sensing modalities, such as vision, audio, motion, and physiological signals, provides a more comprehensive understanding of the environment compared to relying on a single modality. Multi-modal inference leverages these diverse inputs to gain richer contextual awareness, enabling more robust classifications. Our proposed dual-core fusion approach brings advanced multi-modal intelligence to highly constrained microcontrollers. By distributing specialized uni-modal models across the dual cores and combining their intermediate outputs, we enable deployable multi-modal inference on ultra-low power devices and increase classification accuracy by 12.27% compared to uni-modal image classification. Our parallel processing approach demonstrates that multi-model, multi-modal analysis is achievable even on highly constrained microcontrollers, opening new possibilities for privacy-preserving edge intelligence.

Instead of attempting to train a single large multi-modal neural network, our architecture dedicates separate models optimized for each data modality, helping to improve accuracy. The core processing visual imagery deploys a two-dimensional convolutional neural network architecture well-suited for extracting spatial hierarchies and local correlations in image data. In parallel, the second core utilizes a one-dimensional convolutional neural network to model sequential dependencies present in audio signals. Employing modality-specific architectures significantly improves learning efficiency and overall task performance compared to using a generic network architecture. Furthermore, our approach of running these specialized uni-modal networks in parallel across the dual cores reduces processing time by 48ms compared with sequential execution and avoids redundant computation.

While model parallelisation has been extensively

studied in cloud and mobile computing environments [12], our work addresses fundamentally different challenges unique to multi-modal fusion on ultra-constrained microcontrollers. Traditional approaches assume large memory resources and shared memory architectures which not available in microcontrollers where the total memory is often limited to under 2MB with physically isolated memory spaces between cores. Rather than focusing on workload distribution or pipeline scheduling as in traditional approaches [2], we must carefully consider every byte of memory usage and each instance of cross-core communication. Our work demonstrates that effective multi-modal fusion is achievable under these extreme constraints through careful feature compression, strategic fusion point selection, and optimized cross-core communication methods such as the utilisation of additional dense layers to compress features before communication.

Dual-core processors in microcontrollers offer unique opportunities for parallel processing in resource-constrained environments, allowing specialized tasks to be executed simultaneously while maintaining low power consumption and small form factors. While our work focuses on efficient multi-modal fusion through parallel processing across dual cores, other work has explored different approaches to dual-core architectures for machine learning at the edge. Zhao et al. [17] proposed a heterogeneous dual-core architecture optimized for different types of CNN layers, where one core handles regular convolution and another handles depthwise convolution. Their approach demonstrates how core specialization can improve processing efficiency for specific layer types. In contrast, our work addresses the distinct challenge of fusing multiple modalities (audio and visual) in parallel, requiring careful consideration of cross-core feature sharing and memory constraints.

Our implementation demonstrates several key advantages for real-world applications including privacy preservation as by processing data entirely at the edge, sensitive information never leaves the device, reduced latency to enable real-time applications and deployment flexibility as a total size of 258KB enables wide hardware compatibility. The demonstrated capabilities enable new applications across multiple domains. In healthcare monitoring, wearable devices could combine audiovisual and physiological data for more accurate health status assessment. Industrial IoT systems could fuse multiple sensor streams for improved equipment fault detection and smart home devices could better understand complex domestic situations by combining visual and acoustic analysis.

While our parallel feature fusion approach demon-

strates clear benefits, it necessitates careful consideration of resource allocation. The need to share intermediate feature embeddings across cores increases overall memory usage, as feature vectors must be stored both in the original core's memory space and again after transmission to the secondary core where fusion occurs. This redundant buffer overhead stems from the memory architecture and lack of unified shared memory access between cores on many microcontroller systems.

However, this memory cost remains relatively low compared to the full activation tensors, and represents an acceptable trade-off to enable efficient multi-model fusion without direct tensor sharing. By keeping the dense embedding layers compact, the memory overhead remains manageable even on highly constrained microcontrollers. The small increase in memory utilization enables substantial accuracy benefits through specialized modality-specific inferences.

Our implementation focuses on dual-core architectures, although the proposed methodology can be extended to microcontrollers with more than two cores. Each additional core could process another modality or task in parallel, with their feature embeddings being concatenated into the joint feature vector before final classification. This scalability enables even greater multi-modal fusion as hardware capabilities expand, though careful consideration would be needed to manage the increased memory requirements of transmitting multiple feature vectors between cores.

As on-device memory capacities continue growing, the current limitations around feature sharing between cores will likely diminish. This evolution could enable more sophisticated fusion strategies while maintaining the benefits of parallel processing. The demonstrated ability to perform sophisticated multi-modal analysis on constrained devices has significant implications for pervasive computing. As IoT devices become increasingly prevalent, the need for intelligent, privacy-preserving edge processing grows. Our approach shows that even highly constrained devices can perform complex sensor fusion tasks previously reserved for cloud computing or specialized hardware. Furthermore, the current implementation focuses on dual-modality fusion, but real-world scenarios might benefit from incorporating additional sensing modes. Future work could explore scaling to three or more modalities while maintaining efficient resource utilization.

Overall, on-device multi-modal parallel feature fusion overcomes the latency, bandwidth, and privacy limitations of cloud-based inference, presenting opportunities for novel applications beyond scene recognition. These possibilities highlight the potential of

concurrently running multiple specialized architectures across multiple cores, enabling greater modelling flexibility for each modality without the need for sequential execution. This approach leads to improved feature extraction and higher accuracy compared to a uniform model architecture design while also reducing latency for real-time processing.

Conclusion

This work demonstrates the feasibility and effectiveness of parallel feature fusion for multi-modal scene recognition on dual-core microcontrollers. By leveraging the Portenta H7's dual-core architecture, we successfully executed specialized models for visual and audio processing simultaneously, achieving a 12.27% increase in classification accuracy compared to single-modality approaches while reducing latency by 48ms through parallel processing.

Our implementation overcomes several key challenges in edge computing. The parallel processing approach, coupled with efficient feature fusion, enables sophisticated multi-modal analysis within the strict resource constraints of microcontrollers. The total system footprint of 258KB, well within the 2MB limit of typical microcontrollers, demonstrates that complex AI tasks can be accomplished on constrained hardware without sacrificing performance. On the other hand, the parallel execution allows for more efficient utilization of the available computing resources on the microcontroller. Future research could explore the extension to other modalities and the exploration of alternative fusion techniques to further improve performance.

REFERENCES

- [1] Colby R. Banbury et al. "Benchmarking TinyML Systems: Challenges and Direction". In: (Mar. 2020). URL: <https://arxiv.org/abs/2003.04821v4>.
- [2] Tal Ben-Nun and Torsten Hoefler. "Demystifying Parallel and Distributed Deep Learning". In: *ACM Computing Surveys (CSUR)* 52.4 (Aug. 2019). ISSN: 15577341. DOI: [10.1145/3320060](https://doi.org/10.1145/3320060). URL: <https://dl.acm.org/doi/10.1145/3320060>.
- [3] Farshid Hassani Bijarbooneh et al. "Cloud-Assisted Data Fusion and Sensor Selection for Internet of Things". In: *IEEE Internet of Things Journal* 3.3 (June 2016), pp. 257–268. ISSN: 23274662. DOI: [10.1109/JIOT.2015.2502182](https://doi.org/10.1109/JIOT.2015.2502182).
- [4] Jordan J. Bird et al. "Look and Listen: A Multimodality Late Fusion Approach to Scene Classification for Autonomous Machines". In: *IEEE International Conference on Intelligent Robots and Systems* (July 2020), pp. 10380–10385. ISSN: 21530866. DOI: [10.1109/IROS45743.2020.9341557](https://doi.org/10.1109/IROS45743.2020.9341557). URL: <https://arxiv.org/abs/2007.10175v1>.
- [5] Luigi Capogrosso et al. "A Machine Learning-oriented Survey on Tiny Machine Learning". In: *IEEE Access* 12 (Sept. 2023), pp. 23406–23426. ISSN: 21693536. DOI: [10.1109/ACCESS.2024.3365349](https://doi.org/10.1109/ACCESS.2024.3365349). URL: <https://arxiv.org/abs/2309.11932v2>.
- [6] Yu Cheng et al. "Model Compression and Acceleration for Deep Neural Networks: The Principles, Progress, and Challenges". In: *IEEE Signal Processing Magazine* 35.1 (Jan. 2018), pp. 126–136. ISSN: 10535888. DOI: [10.1109/MSP.2017.2765695](https://doi.org/10.1109/MSP.2017.2765695).
- [7] Michael Gibbs and Eiman Kanjo. "Realising the Power of Edge Intelligence: Addressing the Challenges in AI and tinyML Applications for Edge Computing". In: *2023 IEEE International Conference on Edge Computing and Communications (EDGE)*. Institute of Electrical and Electronics Engineers (IEEE), Sept. 2023, pp. 337–343. DOI: [10.1109/edge60047.2023.00056](https://doi.org/10.1109/edge60047.2023.00056).
- [8] Michael Gibbs, Kieran Woodward, and Eiman Kanjo. "Combining Multiple tinyML Models for Multimodal Context-Aware Stress Recognition on Constrained Microcontrollers". In: *IEEE Micro* (Nov. 2023), pp. 1–9. ISSN: 0272-1732. DOI: [10.1109/MM.2023.3329218](https://doi.org/10.1109/MM.2023.3329218).
- [9] Raffaele Gravina et al. "Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges". In: *Information Fusion* 35 (May 2017), pp. 68–80. ISSN: 1566-2535. DOI: [10.1016/J.INFFUS.2016.09.005](https://doi.org/10.1016/J.INFFUS.2016.09.005).
- [10] Caner Hazirbas et al. "FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10111 LNCS (2017), pp. 213–228. ISSN: 16113349. DOI: [10.1007/978-3-319-54181-5_14](https://doi.org/10.1007/978-3-319-54181-5_14). URL: https://link.springer.com/chapter/10.1007/978-3-319-54181-5_14.
- [11] Dana Lahat, Tulay Adali, and Christian Jutten. "Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects". In: *Proceedings of the IEEE* 103.9 (Sept. 2015), pp. 1449–1477. ISSN: 15582256. DOI: [10.1109/JPROC.2015.2460697](https://doi.org/10.1109/JPROC.2015.2460697).
- [12] Nicholas D. Lane et al. "Squeezing Deep Learning into Mobile and Embedded Devices". In:

- IEEE Pervasive Computing* 16.3 (2017), pp. 82–88. ISSN: 15361268. DOI: [10.1109/MPRV.2017.2940968](https://doi.org/10.1109/MPRV.2017.2940968).
- [13] Tong Meng et al. “A survey on machine learning for data fusion”. In: *Information Fusion* 57 (May 2020), pp. 115–129. ISSN: 1566-2535. DOI: [10.1016/J.INFFUS.2019.12.001](https://doi.org/10.1016/J.INFFUS.2019.12.001).
- [14] Maciej Pawłowski, Anna Wróblewska, and Sylwia Sysko-Romańczuk. “Effective Techniques for Multimodal Data Fusion: A Comparative Analysis”. In: *Sensors 2023, Vol. 23, Page 2381* 23.5 (Feb. 2023), p. 2381. ISSN: 1424-8220. DOI: [10.3390/S23052381](https://doi.org/10.3390/S23052381). URL: <https://www.mdpi.com/1424-8220/23/5/2381/htm%20https://www.mdpi.com/1424-8220/23/5/2381>.
- [15] Tuomo Sipola et al. “Artificial Intelligence in the IoT Era: A Review of Edge AI Hardware and Software”. In: *Conference of Open Innovation Association, FRUCT 2022-April* (2022), pp. 320–331. ISSN: 23057254. DOI: [10.23919/FRUCT54823.2022.9770931](https://doi.org/10.23919/FRUCT54823.2022.9770931).
- [16] Cees G.M. Snoek, Marcel Worring, and Arnold W.M. Smeulders. “Early versus late fusion in semantic video analysis”. In: *Proceedings of the 13th ACM International Conference on Multimedia, MM 2005* (2005), pp. 399–402. DOI: [10.1145/1101149.1101236](https://doi.org/10.1145/1101149.1101236). URL: <https://dl.acm.org/doi/10.1145/1101149.1101236>.
- [17] Tiandong Zhao et al. “Heterogeneous Dual-Core Overlay Processor for Light-Weight CNNs”. In: *Proceedings - 29th IEEE International Symposium on Field-Programmable Custom Computing Machines, FCCM 2021* (May 2021), p. 264. DOI: [10.1109/FCCM51124.2021.00051](https://doi.org/10.1109/FCCM51124.2021.00051).
- [18] Zhi Zhou et al. “Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing”. In: *Proceedings of the IEEE* 107.8 (Aug. 2019), pp. 1738–1762. ISSN: 15582256. DOI: [10.1109/JPROC.2019.2918951](https://doi.org/10.1109/JPROC.2019.2918951).

is a recent winner of the Top 50 Women in Engineering by the Women in Engineering Society. She is the NTU-Turing Network Academic lead funded by Alan Turing Institute. She received her PhD in computer Science from Abertay University of Dundee, UK. Contact her at eiman.kanjo@ntu.ac.uk

Kieran Woodward is a Senior Research Fellow in the Smart Sensing Lab, Nottingham Trent University, UK. His current research interests include tinyML, pervasive technologies, and on-device AI optimisation. Kieran received a PhD in Pervasive Computing and Data Science from Nottingham Trent University in 2021. Contact him at kieran.woodward@ntu.ac.uk

Eiman Kanjo is a Professor of Pervasive Sensing and the head of the Smart Sensing lab at Nottingham Trent University. She is also a visiting Professor at Imperial College London. She conducts research in Mobile Sensing, wearable, tinyML and Edge Computing. She