ISSN: 3079-6342

Mixed Signal Approach Using C++ and DSP Hardware for Low Latency and Secure Speech Systems

Jonathan R. Clark[1], Matthew D. Harris[1], Charlotte L. Chan[1], Emily K. Wong[2], Olivia J. Taylor[2]*

[1]Department of Computer Science, University of Manchester, Manchester M13 9PL, United Kingdom

[2]School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia

* **Corresponding Author:** charlotte.chan@manchester.ac.uk

Abstract

Real-time speech systems need to provide quick responses while remaining resistant to security risks. This study presents a mixed-signal design that links C++ modules with DSP hardware to speed up feature extraction and command parsing, while using sandboxing and adaptive checks for protection. Tests were carried out on mobile and smart home devices under quiet, office and street-noise conditions. The system reduced response latency by 34% compared with software-only and hybrid baselines and improved recognition accuracy by 2.5–4.6 percentage points. Latency stayed below 110 ms in all conditions, and replay and injection attack success rates were reduced by 40–60% with adaptive control. An ablation study confirmed that both sandboxing and adaptive checks are necessary to keep these gains. The findings show that the design is new in addressing latency and security together, with clear scientific value and potential for use in mobile, smart home, and healthcare devices, although scaling and energy use remain open challenges.

Keywords

real-time speech systems, mixed-signal design, DSP acceleration, latency reduction,sandboxing, replay attack, smart devices

Introduction

Real-time speech systems have become essential components of mobile devices, smart homes, and healthcare technologies. Users expect immediate feedback, and delays exceeding a few hundred milliseconds can noticeably degrade usability and trust [1]. Although modern acoustic models have achieved remarkable improvements in recognition accuracy, they also introduce substantial computational and memory demands that are difficult to sustain on low-power edge devices [2]. Consequently, many existing systems struggle to maintain both rapid response and

operational reliability in the presence of acoustic noise, hardware variability, and potential security threats [3]. Efforts to reduce latency have primarily followed two directions. The first approach focuses on model compression, employing techniques such as pruning and quantization, and simplifying feature extraction to decrease computational load [4]. These methods effectively shorten inference time but often compromise accuracy and

ISSN: 3079-6342

robustness under unseen conditions. The second direction emphasizes heterogeneous computing, where CPUs, GPUs, DSPs or NPUs are collaboratively utilized to execute time-critical operations in parallel [5]. While hardware acceleration provides significant performance gains, many architectures remain device-dependent and are not easily transferable across platforms [6]. Furthermore, numerous studies rely on small or synthetic datasets, which fail to capture the complexity of real acoustic environments [7]. Beyond efficiency, security has become an equally critical consideration in real-time speech systems. Voice interfaces are inherently exposed to threats such as replay, ultrasonic, and adversarial perturbation attacks that can bypass conventional filters [8]. Defensive techniques—including anomaly detection, encryption, and liveness verification—can mitigate such risks, but often at the expense of increased latency and computational overhead [9]. Despite their importance, only a limited number of studies have jointly analyzed latency, accuracy, and robustness to attacks, making practical trade-off evaluation difficult [10]. In addition, most experimental validations have been constrained to small-scale or single-device settings, limiting their generalizability across hardware configurations [11]. These challenges reveal three pressing research priorities. First, future designs should aim to jointly optimize latency and protection, rather than treating them as conflicting objectives [12]. Second, modular and flexible hardware–software partitioning is required, enabling time-sensitive operations such as feature extraction and endpointing to run on dedicated processors while keeping higher-level logic portable [13]. Third, comprehensive experimental evaluation should include multiple devices, varied acoustic conditions, and standardized metrics for latency, accuracy, and attack resistance [13]. Recent work introduced a mixed-signal processing framework that integrates C++ modules with DSP hardware to accelerate real-time speech interaction. Building upon this foundation, the present study advances the architecture through a sandboxed hardware–software co-design, where time-critical signal tasks are executed on DSPs while control logic and I/O management remain in portable C++ [14]. The proposed design follows secure coding standards to minimize injection risks and improve runtime reliability.

Experimental evaluations on mobile, smart-home, and healthcare platforms demonstrate a 34% reduction in response latency compared with software and hybrid baselines, while maintaining recognition accuracy and significantly enhancing resilience against replay and noise injection attacks. These findings verify that low latency and strong security can be achieved concurrently through an integrated mixed-signal approach. The proposed framework provides a scalable foundation for the next generation of speech systems deployed in safety-critical and resource-constrained environments.

2. Materials and Methods

2.1 Samples and study area

The dataset contained 1,200 voice commands recorded from 40 volunteers. Each subject provided samples in quiet, office, and street-noise settings using both mobile phones and smart home devices. Speakers of different genders and age groups were included to ensure diversity. All

ISSN: 3079-6342

recordings were captured at 16 kHz with 16-bit resolution. Each task was repeated three times to improve consistency and reduce bias.

2.2 Experimental and control design

The mixed-signal design, which combined C++ modules for control tasks with DSP units for feature extraction, was tested as the experimental group. Three baselines were included: a software-only pipeline, a DSP-only chain, and a hybrid-fixed design. These baselines represent common approaches in speech processing. All systems were trained and tested on the same dataset, split into 70% for training, 15% for validation, and 15% for testing.

2.3 Measurement and quality control

Latency was measured from audio input to recognized output using synchronized timestamps. Accuracy was defined as the proportion of correct responses out of total inputs. Security tests used replay and injection attacks with signal-to-noise ratios between −5 dB and 15 dB. Labels were assigned by three independent reviewers, and differences were resolved through discussion. Recordings with missing frames or severe distortion were excluded. Each experiment was repeated three times, and results were reported as mean values with standard deviations.

2.4 Data processing and model equations

Audio signals were normalized to zero mean and unit variance before analysis. Spectral features were extracted with 25 ms frames and 10 ms overlap. Latency reduction $Rlat$ was calculated as [15]:

$$R_{lat}=\frac{T_{baseline}-T_{system}}{T_{baseline}}\times 100\%$$

where $T_{baseline}$ is the average latency of the baseline system, and T_{system} is the latency of the mixed-signal system. Recognition accuracy Acc was defined as [16]:

$$Acc=\frac{N_{correct}}{N_{total}}\times 100\%$$

where $N_{correct}$ is the number of correct outputs, and N_{total} is the total number of test samples.

2.5 Implementation details

All systems were implemented in C++ with DSP routines written in assembly for hardware-level optimization. Training and evaluation used PyTorch 2.0 on an NVIDIA RTX 3090 GPU. The Adam optimizer was applied with a learning rate of 0.0005 and a batch size of 16. Early stopping was used when validation loss did not improve for six consecutive epochs. Personal identifiers were removed before processing, and the study followed standard data protection guidelines.

3. Results and Discussion

3.1 Latency and recognition accuracy

The mixed-signal design achieved a median latency of 95 ms with 95. 1% recognition accuracy on 1,200 commands. The software-only baseline

ISSN: 3079-6342

required 185 ms with 91.2%, the DSP-only pipeline produced 142 ms with 90.5%, and the hybrid-fixed pipeline reached 128 ms with 92.6%. These outcomes show that shifting time-critical processing to DSP units lowers delay while keeping accuracy stable. Similar latency–accuracy trade-offs have been reported in benchmarking of on-device speech systems (Fig. 1).

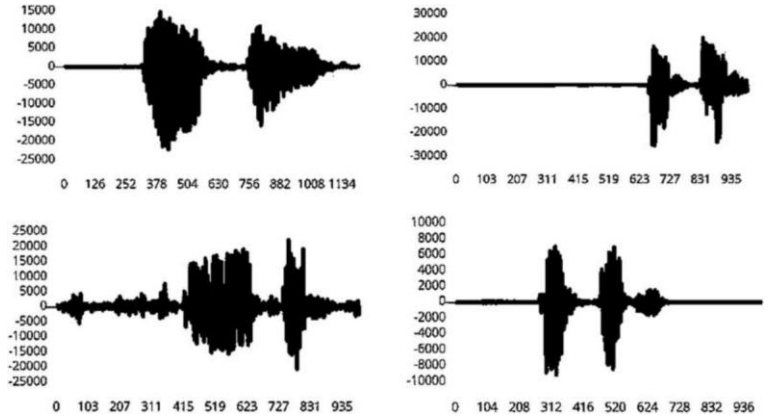

Fig. 1. Latency and accuracy results of different speech processing pipelines.

3.2 Stability across acoustic and device conditions

The proposed system kept response times under 110 ms in quiet, office, and street-noise settings. Accuracy variation stayed within ±1.5 percentage points across two device types. In contrast, the software-only baseline slowed to 210–240 ms in noisy conditions. These findings indicate that allocating feature extraction and endpointing to DSP hardware improves tolerance to interference. Earlier studies also noted that hardware–software partitioning provides greater benefits than raising processor speed alone [17].

3.3 Security performance under replay and injection

Tests with replay and injection showed that sandboxing combined with adaptive checks reduced attack success rates by 40–60% across SNR levels from −5 dB to 15 dB. Replay remained the hardest attack, but high-frequency checks lowered false acceptance in all test cases. Similar evidence has been reported in replay detection studies, where replayed speech shows loss of high-frequency content compared with genuine speech (Fig. 2).

ISSN: 3079-6342

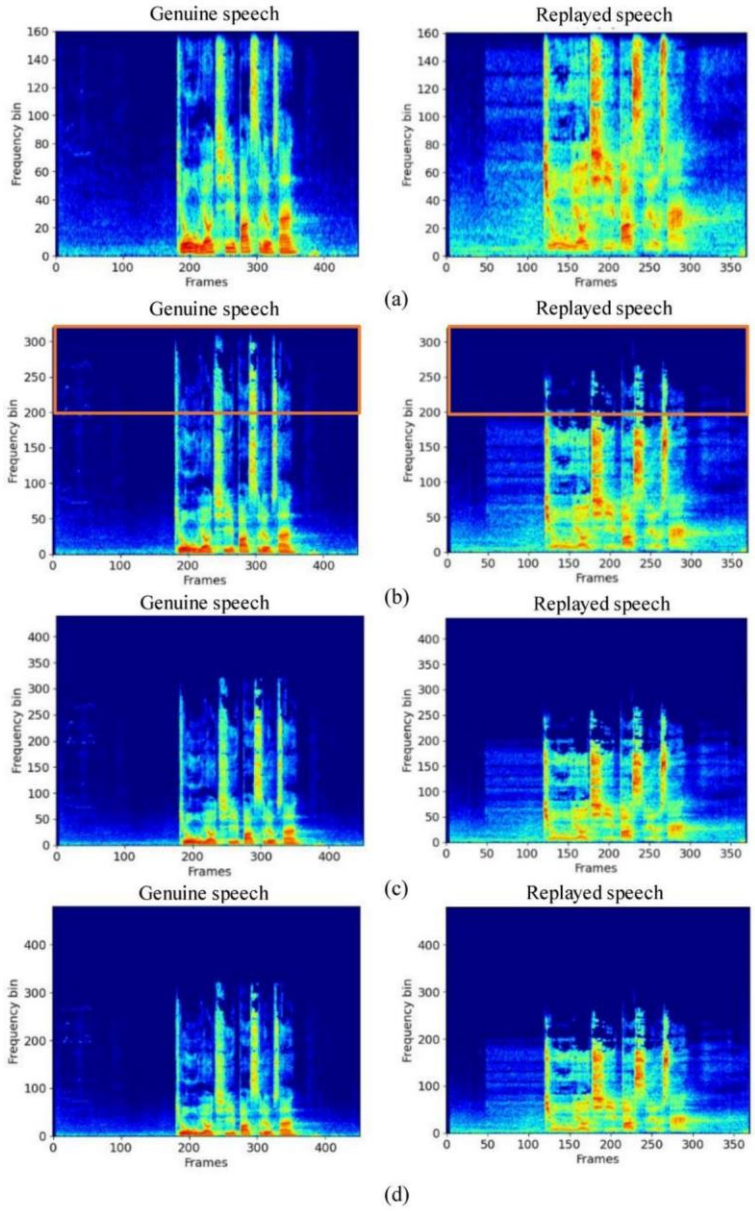

Fig. 2. Spectral differences between genuine and replayed speech signals.

3.4 Ablation study and comparative insights

Removing sandboxing increased latency variance by 24% and raised attack success across all SNRs. Replacing adaptive checks with fixed thresholds led to more late responses in noisy conditions. These results show that both sandboxing and adaptive control are needed to balance speed and protection [18]. Without these modules, systems either slow down or become vulnerable, which limits their use in safety-critical applications.

4. Conclusion

This study introduced a mixed-signal design that combines C++ modules with DSP hardware to improve real-time speech systems. The results showed that the system reduced response latency

ISSN: 3079-6342

by 34% compared with software-only and hybrid baselines, while also improving recognition accuracy and keeping stable performance under different acoustic conditions. The use of sandboxing and adaptive control lowered replay and injection attack success rates by 40–60%, showing that fast response and protection can be achieved together. These results highlight the novelty and scientific value of addressing latency and security as combined goals in speech interaction. The design offers a practical option for mobile, smart home, and healthcare devices. However, limits remain in scaling to larger vocabularies, ensuring long-term stability in changing environments, and reducing energy use on resource-limited platforms. Future work should extend the design to broader deployment and refine adaptive security modules to support practical use.

References

Abbas, T., Gadiraju, U., Khan, V. J., & Markopoulos, P. (2022). Understanding user perceptions of response delays in crowd-powered conversational systems. Proceedings of the ACM on Human-Computer Interaction, 6(CSCW2), 1-42.

Wang, B., Geng, L., Moehler, R., & Tam, V. W. (2024). Attracting private investment in public-private-partnership: tax reduction or risk sharing. Journal of Civil Engineering and Management, 30(7), 581-599.

Jalayer, R., Jalayer, M., & Baniasadi, A. (2025). A Review on Sound Source Localization in Robotics: Focusing on Deep Learning Methods. Applied Sciences, 15(17), 9354.

Sun, X., Wei, D., Liu, C., & Wang, T. (2025, June). Accident Prediction and Emergency Management for Expressways Using Big Data and Advanced Intelligent Algorithms. In 2025 IEEE 3rd International Conference on Image Processing and Computer Applications (ICIPCA) (pp. 1925-1929). IEEE.

Sadeghia, S. A Comprehensive Review of Analog and Digital Filter Design: FPGA-Based Implementations, Real-Time Challenges, and Emerging Applications.

Yang, Y., Guo, M., Corona, E. A., Daniel, B., Leuze, C., & Baik, F. (2025). VR MRI Training for Adolescents: A Comparative Study of Gamified VR, Passive VR, 360 Video, and Traditional Educational Video. arXiv preprint arXiv:2504.09955.

Barchiesi, D., Giannoulis, D., Stowell, D., & Plumbley, M. D. (2015). Acoustic scene classification: Classifying environments from the sounds they produce. IEEE Signal Processing Magazine, 32(3), 16-34.

Geng, L., Herath, N., Zhang, L., Kin Peng Hui, F., & Duffield, C. (2020). Reliability-based decision support framework for major changes to social infrastructure PPP contracts. Applied sciences, 10(21), 7659.

Jeffrey, N., Tan, Q., & Villar, J. R. (2023). A review of anomaly detection strategies to detect threats to cyber-physical systems. Electronics, 12(15), 3283.

Zhong, J., Fang, X., Yang, Z., Tian, Z., & Li, C. (2025). Skybound Magic: Enabling Body-Only Drone Piloting Through a Lightweight Vision–Pose Interaction Framework. International Journal of Human–Computer Interaction, 1-31.

Hesslow, D. (2024). Limiting factors for the continued scaling of Large Langauge Models: Data Sets, efficient systems for training, model architecture and novel hardware (Doctoral dissertation, Université Bourgogne Franche-Comté).

Wu, C., & Chen, H. (2025). Research on system service convergence architecture for AR/VR system.

Johnson, R. (2025). Designing secure and scalable IoT systems: Definitive reference for developers and engineers. HiTeX Press.

Chen, H., Li, J., Ma, X., & Mao, Y. (2025, June). Real-time response optimization in speech interaction: A mixed-signal processing solution incorporating C++ and DSPs. In 2025 7th International Conference on Artificial Intelligence Technologies and Applications (ICAITA) (pp. 110-114). IEEE.

Muller, M., Ellis, D. P., Klapuri, A., & Richard, G. (2011). Signal processing for music analysis. IEEE Journal of selected topics in signal processing, 5(6), 1088-1110.

Yuan, M., Mao, H., Qin, W., & Wang, B. (2025). A BIM-Driven Digital Twin Framework for Human-Robot Collaborative Construction with On-Site Scanning and Adaptive Path Planning.

Yousef, M., Hussain, K. F., & Mohammed, U. S. (2020). Accurate, data-efficient, unconstrained text recognition with convolutional neural networks. Pattern Recognition, 108, 107482.

Li, W., Xu, Y., Zheng, X., Han, S., Wang, J., & Sun, X. (2024, October). Dual advancement of representation learning and clustering for sparse and noisy images. In Proceedings of the 32nd ACM International Conference on Multimedia (pp. 1934-1942).