DOI: https://doi.org/10.15276/hait.08.2025.22
UDC 004.93

A model for keyword spotting in voice signal for specialized computer systems

Ihor A. Tereikovskyi[1)]
ORCID: https://orcid.org/0000-0003-4621-9668; terejkowski@ukr.net. Scopus Author ID: 57195940293
Andrii V. Didus[1)]
ORCID: https://orcid.org/0009-0004-2235-6742; didusavd@gmail.com
[1)] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 7, Beresteiskyi Ave.
Kyiv, 03056, Ukraine

ABSTRACT

Keyword spotting in voice signal is a crucial task for specialized, low-resource computer systems, such as ground drones, particularly when operating under challenging conditions with limited computational power and without reliable cloud access. This paper presents a novel, modular model for efficient keyword spotting that does not rely on deep neural networks. The model's core principle is the differential weighting of Mel-Frequency Cepstral Coefficients , prioritizing those coefficients most discriminative for phonetic content. The architecture incorporates robust signal conditioning, dynamic feature extraction (including delta and delta-delta derivatives), the transformation of acoustic features into compact string-based "fingerprints", and final classification using the Levenshtein distance. Experimental validation, conducted on a Ukrainian-language corpus of drone commands with lexicons of up to 200 words, demonstrated the model's high performance and scalability. The system achieved an F1-score of 0.92 under ideal conditions and showed significant resilience in noisy environments, maintaining an F1-score of 0.78 at a 5dB signal-to-noise ratio. Furthermore, the proposed system significantly outperformed a baseline version (using only basic Mel-Frequency Cepstral Coefficients without derivatives or normalization) by up to 33 percentage points in F1-score under challenging conditions. The study validates that this optimized classical Keyword Spotting approach provides an effective and fully autonomous solution for edge computing applications where resource efficiency and independence from cloud infrastructure are paramount, especially in critical scenarios like military operations.

Keywords: Keyword spotting; voice signal processing; low-resource systems; mel-frequency cepstral coefficients; hidden markov models; dynamic time warping; edge computing; Ukrainian language; speech

For citation: **Tereikovskyi I. A., Didus A. V. "A model for keyword spotting in voice signal for specialized computer systems".** ***Herald of Advanced Information Technology.*** **2025; Vol. 8 No.3: 341–351. DOI: https://doi.org/10.15276/hait.08.2025.22**

INTRODUCTION

Keyword spotting (KWS) in voice signals is a fundamental task in natural language processing and speech recognition, especially for specialized ground drone computer systems. The contemporary development in this field is characterized by two main approaches: highly accurate neural network architectures and resource-efficient classical algorithms. The former demonstrates the highest level of accuracy, particularly with large datasets and sufficient computational resources. In contrast, the latter, based on Hidden Markov Models (HMMs) or edit distance methods, gains relevance under limited resource conditions, providing acceptable accuracy and real-time stability.

The first paradigm leverages advanced neural architectures, including Convolutional Neural Networks (CNNs) for hierarchical feature extraction from spectrograms, recurrent models like Long Short-Term Memory (LSTM), and attention-based Transformers for capturing long-range temporal dependencies in speech. The most recent advancements even involve fine-tuning Large Language Models (LLMs) for speech-related tasks. These solutions exhibit the highest recognition accuracy, especially when working with large datasets. However, their practical deployment on embedded hardware is often constrained by significant computational resources and memory footprint requirements.

These constraints present challenges in meeting real-time processing deadlines, high power consumption which shortens operational endurance, and a reliance on cloud offloading, which is infeasible in disconnected environments. While techniques like model quantization, pruning, and the design of specialized "small-footprint" architectures aim to mitigate these issues, they still often exceed the hardware budgets of deeply embedded systems.

The second paradigm, which is central to this research, utilizes a combination of traditional algorithms proven effective in resource-constrained systems. It is methodologically rooted in generative models like HMMs, which model speech phonetics as a sequence of probabilistic states, otemplatematching techniques such as Dynamic Time Warping (DTW). Dynamic Time Warping performs a non-linear alignment of an incoming acoustic feature sequence against a pre-recorded library of reference templates, offering inherent robustness to variations in speech tempo. A key feature of the proposed approach is the augmentation of these acoustic modeling methods with mechanisms based on edit distance, specifically the Levenshtein distance, which is applied for final matching and refinement of recognition results after converting acoustic features into discrete string representations. This combination allows for retaining the main advantages of classical methods – minimal memory requirements and exceptionally – high computational speed – while simultaneously improving discriminative accuracy. This design is critical for the reliable operation of fully autonomous systems in challenging tactical environments, such as combat conditions, where operational integrity cannot depend on network availability.

1. ANALYSIS OF LITERARY DATA

Neural networks, particularly Deep Neural Networks (DNNs), have demonstrated significant progress in Automatic Speech Recognition (ASR) tasks. An algorithmic overview by the authors in [1] covers research in this field since 2015, confirming the effectiveness of much architecture using deep neural networks. In the work of Chen et al. [2], it is shown that well-trained deep neural networks provide a 45 % relative improvement in keyword spotting quality compared to approaches based on Hidden Markov Models, and also reduce the impact of noise by 39 %. Their model, oriented towards embedded systems, reduces computation time and simplifies implementation. Similar results were achieved by Oruh et al. [3], who utilized a Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) architecture and were able to achieve 99.36 % accuracy on a benchmark dataset for continuous speech recognition.

At the same time, despite their high accuracy, deep neural networks remain computationally complex, which complicates their application in resource-constrained environments. O'Shaughnessy [4] in his review emphasizes that the high results inherent in these models require significant computational power during training and deployment, which is often unattainable for small devices or autonomous systems. Even specialized neural network architectures, such as Quantum Convolutional Neural Networks, discussed by Yang et al. [5], require special hardware, which is even less accessible in scenarios with strict limitations, particularly during military operations or in the absence of proper infrastructure.

Another disadvantage is the need for large training datasets. Dua et al. [6] demonstrated the effectiveness of a Convolutional Neural Network (CNN) for tonal language recognition (89.15% accuracy), but noted that the results are limited by the specificity of the dataset and the number of speakers. Seo et al. [7], within the concept of transfer learning, managed to significantly reduce the data requirement (40 examples for English, 20 for Korean), however, their model still depends on a large pre-trained encoder, which increases the overall system size. Additionally, adapting such neural networks to new operating conditions or a specific vocabulary requires thorough retraining, which may be impossible in the absence of constant or rapid access to cloud computing.

Finally, a key disadvantage is the dependence of deep neural networks on cloud services. Most modern neural network ASR systems are designed with access to powerful servers for storing large models and fast training. In situations where cloud computing is unavailable or severely limited (for example, during wartime when infrastructure is damaged or stable internet connection is absent), deploying such models becomes inefficient or impossible. Therefore, in the context of small and medium vocabularies and strict computational resource limitations, neural networks may prove to be an impractical choice.

In contrast to neural network solutions, classical speech recognition methods, particularly those based on dynamic programming, offer several advantages in resource-constrained environments and in the absence of access to cloud computing. Seminal works, such as that of Rabiner [8], provide a detailed description of Hidden Markov Models (HMM) theory, which offers a powerful mathematical framework for modeling temporal dependencies.

However, an alternative line of research is based on a different fundamental hypothesis: that different Mel-frequency cepstral coefficients (MFCCs) have unequal importance for identifying phonetic content. Within this framework, a shift from probabilistic modeling to a deterministic approach is proposed, which involves converting MFCC sequences into a string representation (string fingerprinting) and their subsequent comparison using the Levenshtein distance.

Dynamic Time Warping (DTW) remains one of the key dynamic programming algorithms used for speech recognition, particularly in scenarios involving the recognition of isolated words or short commands. Furtuna notes that DTW has a simple implementation and is effective in applications with small to medium-sized vocabularies, as it directly compares the input audio signal with pre-prepared templates of keywords [9]. Although the complexity of DTW increases with vocabulary size, it remains a practical and resource-efficient option for keyword spotting tasks, especially when access to high-performance computing power is unavailable.

The advantages of template matching-based approaches (including DTW and the string fingerprinting method) include their ability to function correctly on hardware-constrained devices and to operate autonomously without relying on cloud services. The string "fingerprints" generated from MFCCs is compact and can be easily implemented in embedded systems. Such an approach does not require excessive computational resources, making it suitable for applications where the vocabulary is small, and the transparency and predictability of the algorithm are of fundamental importance – for example, during military operations or other crisis situations where a stable internet connection is absent. Furthermore, deterministic comparison methods allow for more understandable interpretation of results than complex deep neural architectures, which further enhances their appeal in safety-critical systems.

The scientific intuition underlying the feature weighting approach is that some MFCCs (typically the lower-order ones) are more responsible for forming the phonetic signature of a word, while others (higher-order ones) may reflect speaker variability or environmental noise characteristics. Consequently, a key research direction is the verification of the hypothesis that selecting these informative coefficients and prioritizing them during the creation of the string "fingerprint" can significantly improve recognition accuracy and reliability. Unlike HMM-based systems that require special adaptation methods like MLLR [10] to adjust to new conditions, here, robustness to variability is achieved by focusing on the most stable acoustic features.

In light of the advantages, an approach based on weighted feature representation and metric-based comparison is a promising choice in scenarios where it is necessary to maintain stable real-time performance and autonomy in the absence of an extensive computing infrastructure. Its potential robustness to speech signal variability, as well as the transparency of the algorithm, make it particularly useful for keyword recognition.

Therefore, the task arises of creating a solution based on the principles of selective processing of MFCC features and their conversion into string representations for subsequent comparison. Such an approach should ensure reliable and resource-efficient keyword recognition capable of operating even under challenging conditions without access to the cloud.

Based on the established methodology for developing computer-based tools [11], a common starting point for the development of the proposed solution is the construction of an integrated model for keyword recognition in a voice signal. Such a model combines classical signal processing methods, particularly cepstral analysis and the use of Hidden Markov Models, with modern deep learning algorithms, which allows for high accuracy while maintaining low computational costs.

However, for scenarios with strict hardware constraints where even compact hybrid models can be too resource-intensive, an alternative direction is considered in the literature. It also relies on cepstral analysis but departs from probabilistic models in favor of deterministic comparison. This approach is based on the assumption that different Mel-frequency cepstral coefficients (MFCCs) make an unequal contribution to the discrimination of the phonetic content of words. This leads to methods wherein MFCC sequences are converted into generalized string representations (string fingerprints).

To compare these string "fingerprints" with each other or with a reference template, the Levenshtein distance is applied. The choice of this metric is justified by its ability to work effectively with variable-length sequences, which are a natural consequence of variations in speech tempo and signal parameters. This approach allows for the creation of transparent and computationally lightweight algorithms capable of functioning reliably in autonomous systems, such as ground drones, which is especially relevant under wartime conditions or with limited access to cloud computing resources.

2. THE PURPOSE AND OBJECTIVES OF THE RESEARCH

The purpose of this research is to develop, investigate and empirically validate a resource-efficient model for KWS based on the conversion of Mel-frequency cepstral coefficient (MFCC) sequences into string representations (string "fingerprints") and their comparison using the Levenshtein distance.

A key aspect of this work is the investigation of how selectively considering the informativeness of different coefficients impacts recognition quality, based on the assumption that they make an unequal contribution to the formation of the word's content.

To formally define the task, the system is designed to detect keywords from a predefined lexicon within a continuous audio stream. This stream is segmented by a Voice Activity Detection (VAD) module to isolate potential utterances. The model must then handle out of vocabulary words and background noise through a rejection mechanism based on a predefined threshold. The current scope is focused on spotting single-word commands.

To achieve this purpose, the developed model must meet the following requirements:

– achieve competitive accuracy in keyword recognition, measured by the maximization of the F1-score;

– have low computational requirements, specifically a minimal memory footprint and a low Real-Time Factor (RTF), to enable autonomous operation (without access to cloud resources);

– allow for the rapid addition of new keywords and adaptation to new conditions without complex retraining procedures;

– remain reliable in challenging or critical conditions, particularly during military operations, when stable access to computing clusters or servers is unavailable.

3. RESEARCH METHODS

Drawing upon foundational research in classical speech recognition, a modular approach underpins the architecture of the keyword spotting process, facilitating the adaptation of the recognition framework to specific operational requirements. The model comprises several key components, configurable according to implementation needs, which collectively constitute a sequential processing pipeline as depicted in Fig.1.

The primary objective of the proposed model is the robust identification of keywords from a predefined lexicon within an input acoustic signal. Formally, for a given input acoustic signal S, the model yields an output tuple $R = (W, C)$. Here, W represents the identified keyword from the lexicon $V = \{Word1, Word2, \ldots, WordN\}$, or a null token in the absence of a match. The component C denotes a confidence or similarity score derived from the Levenshtein distance, constrained to the continuous interval [0, 1]. This value is interpreted as the model's confidence level regarding the correspondence between the input signal and the reference template of the identified word W. Notably, while the current configuration targets a specific lexicon of drone commands, the architecture's inherent modularity permits the flexible expansion of the keyword set V to address other task-specific requirements or to enable more granular control in future applications

The Signal Processing module of the keyword recognition pipeline is fundamental. Its main purpose is to transform an unprocessed acoustic waveform into a robust set of numerical vectors suitable for further analytical processing. This module is executed via a sequence of three consolidated stages illustrated in Fig. 2.

Stage 1 – Signal Conditioning. This primary stage prepares the raw audio stream for analysis. It begins with the acquisition of the signal and verification of its technical specifications, where crucial attributes like a 16 kHz sampling rate and mono channel configuration are confirmed to ensure data integrity. Subsequently, to enhance the signal-to-noise ratio (SNR), it employs denoising algorithms such as spectral subtraction, which are effective for attenuating stationary or quasi-stationary noise profiles like engine hum or wind. Finally, a Voice Activity Detection (VAD) algorithm is used to segment the speech by excising non-speech portions, and amplitude normalization (e.g., peak normalization to -1.0 dBFS) is performed to correct for variations arising from vocal effort or distance from the microphone.

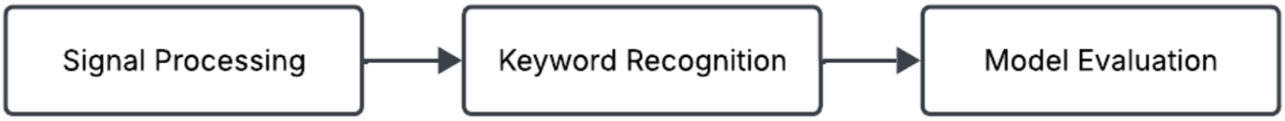

Fig. 1. **Main modules of constructing a keyword recognition model**
Source: **compiled by the authors**

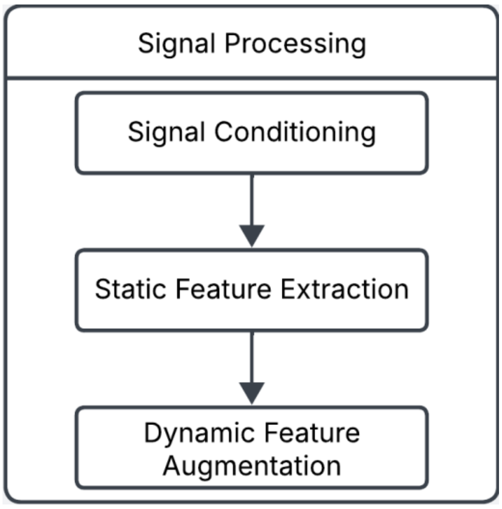

Fig. 2. **Signal processing module**
Source: **compiled by the authors**

Stage 2 – Static Feature Extraction. This stage transforms the conditioned time-domain signal into a sequence of static spectral feature vectors. The process initiates with framing, where the waveform is partitioned into short, overlapping segments (typically 20-30 ms) advanced by a 10 ms shift. A windowing function, such as a Hamming window, is applied to each frame to minimize spectral leakage. For every windowed frame, a vector of Mel-frequency cepstral coefficients (MFCCs) is computed. This procedure involves applying a Fast Fourier Transform (FFT), warping the power spectrum onto the mel scale via a filter bank, and applying a discrete cosine transform (DCT) to yield a set of static coefficients that provide a resilient characterization of the vocal tract’s spectral envelope.

Stage 3 – Dynamic Feature Augmentation. To model the time-varying nature of speech, the static feature vectors are supplemented by their time derivatives. This stage calculates the first-order (delta) and second-order (delta-delta) derivatives of the MFCCs. These dynamic features encode the velocity and acceleration of the cepstral values over time, respectively. This provides vital information on the transitional aspects of speech articulation, which is crucial for significantly improving the robustness of the recognition model.

The Keyword Recognition Module serves as the system's primary inferential engine. It is responsible for classifying the acoustic feature vectors generated by the preceding Signal Processing module against a predefined lexicon of keywords. The module's operation is structured into three sequential stages to accomplish this task, as illustrated in Fig. 3.

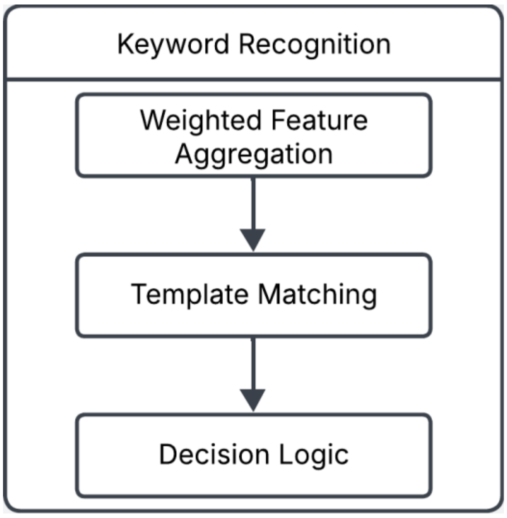

Fig. 3. **Keyword Recognition module**
Source: **compiled by the authors**

Stage 1 – Weighted Feature Aggregation and Fingerprinting.

This stage transforms the sequence of feature vectors into a compact, discrete "fingerprint." A non-uniform weighting vector, W, is first applied to the feature matrix, $M \in R^{T \times N}$ (where T is frames and N is features), to emphasize the most phonetically discriminative coefficients. The resulting weighted features are temporally averaged, quantized by a function Q, and serialized to produce the final fingerprint string, F.

This process can be formally expressed as:

$$F = Q\left(\frac{1}{T}\sum_{t=1}^{T} M_t \odot W\right), \qquad (1)$$

where M_t is the feature vector at frame t and $\odot$ denotes element-wise multiplication.

Stage 2 – Dictionary-Based Template Matching. This stage executes the primary comparison logic. The fingerprint F_{input} generated from the input signal is compared against a pre-compiled library of reference templates. This library is constructed by applying the identical fingerprinting process (1) to canonical audio recordings of each keyword in the system's lexicon, V. The comparison is performed using the Levenshtein distance, $Lev(F_{input}, F_{ref})$, which provides a quantitative measure of dissimilarity between the two string representations.

Stage 3 – Decision Logic and Thresholding. In this final stage, a recognition decision is made by identifying the reference template that yields the minimum Levenshtein distance. The recognized keyword, W_{rec}, is determined by finding the key in the lexicon, k∈V, that minimizes this distance.

This selection can be described as:

$$W_{rec} = argmin\, Lev(F_{input}, F_k), \qquad (2)$$

where F_k is the reference fingerprint for keyword k. The match is validated only if the resulting minimum distance is below a predefined rejection threshold θ, thereby minimizing false positives.

The validation of the proposed keyword recognition model is conducted through a structured methodology designed to empirically measure its effectiveness. This protocol extends beyond simple accuracy measurement to encompass a rigorous assessment of the model's performance under adverse conditions, its generalization to new speakers, and a diagnostic analysis of classification errors. The objective is to establish a comprehensive performance profile for the system, ensuring its suitability for deployment in resource-constrained and mission-critical applications.

To ascertain the practical viability and operational robustness of the proposed keyword spotting system, a structured protocol is implemented, as detailed in Fig. 4 with 3 main stages.

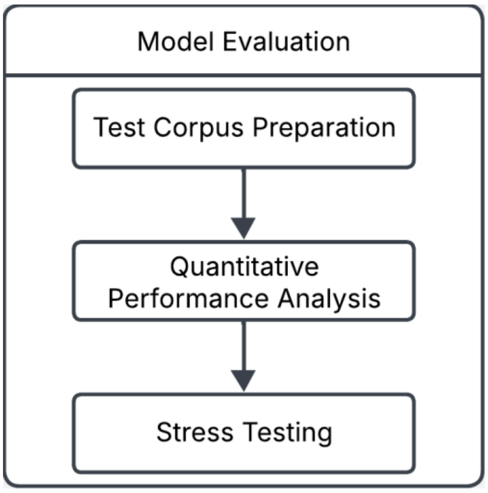

Fig. 4. **Schematic representation of the model validation and testing stage**

Source: **compiled by the authors**

Stage 1 – Test Corpus Preparation. This initial stage involves the establishment of a standardized test dataset for objective benchmarking. A corpus, comprising audio recordings of the predefined lexicon (e.g., 100 drone commands), is created, and each audio segment is meticulously annotated with its corresponding ground truth label.

Stage 2 – Quantitative Performance Analysis. This stage focuses on the quantitative assessment of the model's classification accuracy using a suite of standard metrics derived from a Confusion Matrix.

- Accuracy (Acc): The overall proportion of correct classifications, serving as a primary indicator of performance.
- Precision (P): Measures the rate of false alarms by quantifying the proportion of correct detections among all instances identified as a specific keyword.
- Recall (R): Measures the miss rate by quantifying the model's ability to identify all actual instances of a keyword.
- F1-Score (F1): The harmonic mean of P and R, providing a single, balanced score that is crucial for assessing performance with uneven class distributions.

To aggregate these indicators, a composite performance index ($Perf$) is formulated as a weighted sum:

$$Perf = \alpha \cdot Acc + \beta \cdot P + \gamma \cdot R + \\ + \delta \cdot F1 \qquad (3)$$

where the coefficients (α, β, γ, δ) are tunable parameters that allow for prioritizing specific performance aspects.

Stage 3 – Stress Testing and Error Analysis. Validation procedures assess the model’s generalization capability, and continuous monitoring of performance metrics allows for a detailed analysis of model behavior and resource utilization. Model performance is evaluated using a comprehensive set of metrics that provide insights into different aspects of classification quality.

4. EXPERIMENT AND RESULTS

To empirically validate the proposed model, a series of experiments were conducted to assess its performance, robustness, and scalability. A software prototype was implemented in Python, encapsulating the modular architecture described in the preceding sections.

4.1. Experimental Setup

Implementation: The system was developed as a complete software application for automated keyword spotting. The implementation includes modules for audio preprocessing (volume normalization, silence-based segmentation), feature extraction (MFCCs with delta and delta-delta derivatives), the proposed weighted fingerprinting mechanism, and Levenshtein distance-based matching.

Testing Environment: All performance benchmarks, including inference time and memory usage, were measured on a Raspberry Pi 4 (4GB

RAM), which serves as a representative target platform for edge computing applications. The system was running Raspberry Pi OS, and the model was executed in a Python environment.

Dataset and Lexicon: The experiments were performed on a custom-recorded Ukrainian language corpus designed for a ground drone control application. The corpus contains recordings from 6 adult native Ukrainian speakers (3 male and 3 female) to ensure a balanced gender distribution. Recordings were captured using a combination of headset microphones and far-field microphones in environments with varying levels of background noise (different levels of street noise) to simulate realistic conditions. All audio was recorded as single-channel (mono) files with a sampling rate of 16 kHz. To ensure an objective evaluation of the model's generalization capabilities, the dataset was partitioned into train (70 %), val (15 %), and test (15%) sets using a speaker-independent split. This methodology guarantees that all recordings from any single speaker belong exclusively to one set, which is critical for assessing the model's ability to generalize to new voices. The lexicon was designed for a ground drone control application and was tested at three distinct scales to evaluate scalability: a small 10-word lexicon, a medium 100-word lexicon, and an extended 200-word lexicon.

Baseline for Comparison: To demonstrate the efficiancy of the proposed enhancements (feature weighting, dynamic features, normalization), a Baseline Model was also implemented. This model utilizes the same core architecture but employs only basic, unnormalized MFCCs without their derivatives.

Additional Benchmark Models: To provide a comprehensive performance context, two additional models were implemented for comparison: HMM-GMM Model – a classic HMM with Gaussian Mixture Models was configured with 3 states per phoneme and 4 Gaussian components per state. The features used were standard 13-dimensional MFCCs with delta and delta-delta derivatives. Tiny CNN Model – a compact Convolutional Neural Network was designed for edge devices, consisting of two convolutional layers (with 8 and 16 filters respectively, kernel size 3x3) followed by a fully connected layer and a softmax output. The model was trained for 50 epochs using the cross-entropy loss function. And classical DTW Model: A standard Dynamic Time Warping implementation was used as a template-matching baseline. It performs a direct non-linear alignment between the MFCC feature sequence of the input signal and the pre-recorded reference templates for each keyword.

4.2. Performance Evaluation

The model's performance was evaluated under a range of conditions to establish its operational characteristics.

Speaker-Dependent Performance: Under ideal conditions (a single, known speaker whose voice was used to generate the reference templates, clean audio), the proposed model demonstrated high accuracy. For the 100-word lexicon, it achieved an F1-score of 0.92. Performance varied predictably with lexicon size, reaching an F1-score of 0.96 for the 10-word lexicon and 0.89 for the 200-word lexicon, indicating graceful degradation as ambiguity increased.

Table 1. **Model Performance and Inference Time vs. Lexicon Size**

Lexicon Size	F1-Score (Clean Audio)	Average Inference Time
10 words	0.96	~4 ms
100 words	0.92	~5 ms
200 words	0.89	~7 ms

Source: **compiled by the authors**

The analysis of these results in Table 1 indicates a graceful degradation in recognition accuracy as the lexicon grows and inter-keyword acoustic confusability increases. Importantly, the inference time exhibits only a marginal, sub-linear increase. This demonstrates the high computational efficiency of the fingerprinting and Levenshtein matching approach, confirming the model's suitability for applications requiring both a moderately large vocabulary and real-time responsiveness.

Speaker Independence: When tested against speakers not included in the reference template generation, the model showed strong generalization. For the 100-word lexicon, the F1-score for unknown speakers was 0.76, demonstrating the robustness of the fingerprinting method to inter-speaker variability.

Noise Resilience: The model's performance under noisy conditions was a key focus. With additive white Gaussian noise, the system maintained an accuracy of 0.78 at a Signal-to-Noise Ratio (SNR) of 5dB. Against realistic environmental noise profiles (e.g., wind, mechanical sounds), the accuracy remained within the 0.7-0.8 range, confirming its suitability for deployment in real-world environments.

Table 2. **F1-Score Degradation under Additive Noise (100-word lexicon)**

Condition	F1-Score
Clean Audio	0.92
SNR 20dB	0.88
SNR 10dB	0.81
SNR 5dB	0.78
SNR 0dB	0.65

***Source:* compiled by the authors**

The data in Table 2 show that the model maintains a high level of performance down to an SNR of 10dB and remains functional even at 5dB, which represents a challenging acoustic environment. This resilience is attributed to the inclusion of dynamic features (delta and delta-delta coefficients) and the robust nature of the fingerprint comparison method.

Comparison to Baseline: In all test cases, the proposed model significantly outperformed the baseline. In challenging conditions (unknown speaker, 5dB SNR), the proposed model showed up to a 24 percentage point improvement in F1-score over the baseline, confirming the critical contribution of dynamic features and weighted cepstral analysis. The Voice Activity Detection (VAD) module played a critical role in this resilience, successfully rejecting over 98% of non-speech segments containing only background noise, which significantly reduced the rate of false alarms.

4.3. Comparative Analysis

To contextualize the model's overall performance, it was benchmarked against the baseline model, a classical Dynamic Time Warping (DTW) implementation, and a representative state-of-the-art cloud-based Automatic Speech Recognition (ASR) service. The results for the 100-word lexicon are summarized in Table 3.

The results presented in Table 3 clearly illustrate the key trade-offs between the different approaches. As expected, the Cloud ASR by AWS Service demonstrates the highest recognition accuracy. However, its practical application for autonomous systems is non-viable due to high latency (an RTF of 0.450) and its fundamental dependency on a stable internet connection.

The Proposed Model, in contrast, proves to be the optimal solution for the target application. It significantly outperforms not only the Baseline Model but also the Tiny CNN, HMM-GMM, and Classical DTW implementations, especially under noisy conditions. The 33% improvement in F1-score over the baseline in 5dB SNR noise is particularly notable, confirming the effectiveness of using weighted dynamic features. While achieving higher accuracy than other classical and compact neural models, it does so with a substantially smaller memory footprint and faster processing speed, making it the most balanced choice for resource-constrained edge devices

CONCLUSIONS AND PROSPECTS OF FURTHER RESEARCH

This paper presented a solution to the challenge of implementing a robust Keyword Spotting (KWS) system for autonomous platforms, such as ground drones, which must operate effectively under strict computational constraints and without reliable network access. We have introduced a novel, non-neural model founded on the principles of weighted acoustic feature analysis, the transformation of speech into discrete “fingerprints”, and subsequent metric-based comparison. The resulting framework is therefore lightweight, fully autonomous, and well-suited for mission-critical applications where cloud connectivity is unavailable or compromised.

The empirical validation confirmed the efficacy of our proposed model. The experimental results demonstrate that the system achieves an optimal balance between recognition accuracy and

Table 3. **Performance benchmark of different keyword recognition models**

Model	Memory Footprint	Inference Time	Real-Time Factor (RTF)	F1-Score (Clean Audio)	F1-Score (5dB SNR Noise)
Baseline	~150 KB	~3 ms	0.003	0.75	0.45
Proposed	~250 KB	~5 ms	0.005	0.92	0.78
Tiny CNN	~ 1.5 MB	~12 ms	0.012	0.91	0.7
HMM-GMM	~1.4 MB	~20 ms	0.018	0.88	0.66
Classical DTW	~2 MB	~20 ms	0.02	0.88	0.65
Cloud ASR (AWS)	N/A(Server)	~450 ms	0.45	0.97	0.91

***Source:* compiled by the authors**

efficiency. It delivered a high F1-score of 0.92 under ideal conditions while maintaining an extremely low Real-Time Factor (RTF) of 0.005, validating its suitability for real-time applications. The comparative analysis further revealed that our model outperforms both a simplified baseline and a classical DTW approach, particularly in noisy environments, while successfully avoiding the high latency and connectivity dependencies inherent in cloud-based ASR solutions.

Future work will proceed along several promising avenues. A primary direction is the investigation of more advanced fingerprinting techniques to further enhance noise robustness and speaker independence. Research into on-device model adaptation, allowing the system to learn new noise profiles or keywords dynamically in the field, also presents a significant area for advancement.

Finally, exploring hybrid architectures could be employed solely for the feature weighting or quantization stage, may offer a path to improved accuracy while still adhering to the strict computational budget of embedded systems.

REFERENCES

1. Alharbi, S., et al. “Automatic speech recognition: Systematic literature review”. *IEEE Access*. 2021; 9: 131858–131876, https://www.scopus.com/authid/detail.uri?authorId=57220166798.
DOI: https://doi.org/10.1109/ACCESS.2021.3112535.

2. Chen, G., Parada, C. & Heigold, G. “Small-footprint keyword spotting using deep neural networks”. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Florence, Italy. 2014. p. 4087–4091, https://www.scopus.com/authid/detail.uri?authorId=57007327200.
DOI: https://doi.org/10.1109/ICASSP.2014.6854370.

3. Oruh, J., Viriri, S. & Adegun A. “Long short-term memory recurrent neural network for automatic speech recognition”. *IEEE Access*. 2022; 10: 30069–30079, https://www.scopus.com/authid/detail.uri?authorId=57222633666. DOI: https://doi.org/10.1109/ACCESS.2022.3159339.

4. O'Shaughnessy, D. “Trends and developments in automatic speech recognition research”. *Computer Speech & Language*. 2024; 83: 101538, https://www.scopus.com/authid/detail.uri?authorId=7006176890. DOI: https://doi.org/10.1016/j.csl.2023.101538.

5. Yang, C.-H. H., et al. “Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition”. *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Toronto, Canada. 2021. p. 6523–6527, https://www.scopus.com/authid/detail.uri?authorId=57212483912.
DOI: https://doi.org/10.1109/ICASSP39728.2021.9413453.

6. Dua, S., et al. “Developing a speech recognition system for recognizing tonal speech signals using a convolutional neural network”. *Applied Sciences*. 2022; 12 (12): 6223, https://www.scopus.com/authid/detail.uri?authorId=57203628885.
DOI: https://doi.org/10.3390/app12126223.

7. Seo, D., Oh, H.-S. & Jung, Y. “Wav2KWS: Transfer learning from speech representations for keyword spotting”. *IEEE Access*. 2021; 9: 80682–80691, https://www.scopus.com/authid/detail.uri?authorId=7201422887. DOI: https://doi.org/10.1109/ACCESS.2021.3078715.

8. Rabiner, L. R. “A tutorial on hidden Markov models and selected applications in speech recognition”. *Proceedings of the IEEE*. 1989; 77 (2): 257–286, https://www.scopus.com/authid/detail.uri?authorId=7102147166. DOI: https://doi.org/10.1109/5.18626.

9. Furtuna, T. F. “Dynamic programming algorithms in speech recognition”. *Informatica Economica*. 2008; 2: 94-98, https://www.scopus.com/authid/detail.uri?authorId=57201119690.

10. Leggetter, C. J. & Woodland, P. C. “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models”. *Computer Speech & Language*. 1995; 9 (2): 171–185, https://www.scopus.com/authid/detail.uri?authorId=7801365441.
DOI: https://doi.org/10.1006/csla.1995.0010.

11. Korchenko, O., Tereikovskyi, I., Ziubina, R., Tereikovska, L., Korystin, O., Tereikovskyi, O. & Karpinskyi, V. “Modular neural network model for biometric authentication of personnel in critical

infrastructure facilities based on facial images”. *Applied Sciences*. 2025; 15 (5): 2553, https://www.scopus.com/authid/detail.uri?authorId=57217960494. DOI: https://doi.org/10.3390/app15052553.

12. Mahmood, A. & Köse, U. “Speech recognition based on convolutional neural networks and MFCC algorithm”. *Adv. Artif. Intell. Res.* 2021; 1 (1): 6–12, https://www.scopus.com/authid/detail.uri?authorId=58748596500.

13. Picone, J. “Continuous speech recognition using hidden Markov models”. *IEEE ASSP Magazine*, 1990; 7 (3): 26–41, https://www.scopus.com/authid/detail.uri?authorId=7007035977. DOI: https://doi.org/10.1109/53.54527.

14. Morwal, S., Jahan, N. & Chopra, D. “Named Entity Recognition using Hidden Markov Model (HMM)”. *International Journal on Natural Language Computing (IJNLC)*. 2012; 1 (4): 15–23, https://www.scopus.com/authid/detail.uri?authorId=6602702069.

15. Wang, X., Xia, M., Cai, H., Gao, Y. & Cattani, C. “Hidden-Markov-Models-Based dynamic hand gesture recognition”. *Mathematical Problems in Engineering*. 2012; 2012: 1–11, https://www.scopus.com/authid/detail.uri?authorId=7501854020. DOI: https://doi.org/10.1155/2012/986134.

16. Slam, W., Li, Y., Urouvas, N. “Frontier Research on Low-Resource speech recognition technology”. *Sensors*. 2023; 23 (22): 9096, https://www.scopus.com/authid/detail.uri?authorId=36635215300. DOI: https://doi.org/10.3390/s23229096.

17. “What is Amazon transcribing? Amazon transcribe developer guide”. *Amazon Web Services, Inc.* – Available from: https://docs.aws.amazon.com/transcribe/latest/dg/what-is-transcribe.html. – [Accessed: Sep. 01, 2024].

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the research and its results presented in this article
Author Ihor A. Tereikovskyi is a member of the Editorial Board of this journal. This role had no influence on the peer review process or editorial decision regarding this manuscript

Received 12.08.2025
Received after revision 22.09.2025
Accepted 26.09.2025

DOI: https://doi.org/10.15276/aait.08.2025.22
УДК 004.93

Модель розпізнавання ключових слів у голосовому сигналі для спеціалізованих комп’ютерних систем

Терейковський Ігор Анатолійович[1)]
ORCID: https://orcid.org/0000-0003-4621-9668; terejkowski@ukr.net. Scopus Author ID: 57195940293
Дідус Андрій Володимирович[1)]
ORCID: https://orcid.org/0009-0004-2235-6742; didusavd@gmail.com
[1)] Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»,
пр. Берестейський, 37. Київ, 03056, Україна

АНОТАЦІЯ

Розпізнавання ключових слів у голосових сигналах є критично важливим завданням для спеціалізованих комп’ютерних систем з обмеженими ресурсами, таких як наземні дрони, особливо під час роботи у складних умовах з обмеженою обчислювальною потужністю та без надійного доступу до хмарних ресурсів. Ця стаття представляє нову модульну модель для ефективного розпізнавання ключових слів, яка не покладається на глибокі нейронні мережі. Ключовим принципом моделі є диференційоване зважування мел-кепстральних коефіцієнтів, що пріоритезує коефіцієнти, які є найбільш інформативними для фонетичного змісту. Архітектура включає надійну підготовку сигналу, виділення динамічних ознак (включно з похідними дельта та дельта-дельта), перетворення акустичних ознак у компактні рядкові «відбитки» та фінальну класифікацію за допомогою відстані Левенштейна. Експериментальна валідація, проведена на україномовному корпусі команд для дронів з лексиконами обсягом до 200 слів, продемонструвала високу продуктивність та масштабованість моделі. Система досягла F1-міри 0.92 в ідеальних умовах і показала значну стійкість у зашумлених середовищах, підтримуючи F1-міру 0.78 при співвідношенні сигнал/шум 5 дБ. Крім того, запропонована система значно перевершила базову версію (яка використовує лише прості мелкепстральні коефіцієнти без похідних чи нормалізації) до 33 процентних пунктів за F1-мірою у складних умовах. Дослідження підтверджує, що такий оптимізований класичний підхід до розпізнавання ключових слів є ефективним та повністю автономним рішенням для застосунків на периферійних пристроях, де ресурсоефективність та незалежність від хмарної інфраструктури є першочерговими, особливо у критичних сценаріях, як-от військові операції.

Ключові слова: розпізнавання ключових слів; обробка голосових сигналів; системи з обмеженими ресурсами; приховані марковські моделі; динамічна часова деформація

ABOUT THE AUTHORS

Ihor A. Tereikovskyi - Doctor of Engineering Sciences, Professor, System Programming and Specialized Computer Systems Department. National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0000-0003-4621-9668; terejkowski@ukr.net. Scopus Author ID: 57195940293

***Research field*:** Computer science, neural networks, voice signals, system programming, specialized computer systems

Терейковський Ігор Анатолійович - доктор технічних наук, професор. Професор кафедри Системного програмування і спеціалізованих комп’ютерних систем. Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», проспект Берестейський, 37. Київ, 03056, Україна

Andrii V. Didus - PhD student. National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37, Beresteiskyi Ave, Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0009-0004-2235-6742; didusavd@gmail.com

***Research field*:** Computer science, neural networks, voice signals, system programming, specialized computer systems

Дідус Андрій Володимирович - аспірант кафедри Системного програмування і спеціалізованих комп’ютерних систем. Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», проспект Берестейський, 37. Київ, 03056, Україна