applied sciences

MDPI

Article

Front-End of Vehicle-Embedded Speech Recognition for Voice-Driven Multi-UAVs Control

Jeong-Sik Park [1,*] and Hyeong-Ju Na [2]

[1] Department of English Linguistics and Language Technology, Hankuk University of Foreign Studies, Seoul 02450, Korea
[2] Department of English Linguistics, Hankuk University of Foreign Studies, Seoul 02450, Korea; skgudwn1994@hufs.ac.kr
* Correspondence: parkjs@hufs.ac.kr; Tel.: +82-02-2173-8814

Received: 29 August 2020; Accepted: 29 September 2020; Published: 30 September 2020

Featured Application: This research can be applied to voice-driven control of multiple devices with device-embedded speech recognition. Such systems require efficient front-end processing, including noise reduction and voice trigger.

Abstract: For reliable speech recognition, it is necessary to handle the usage environments. In this study, we target voice-driven multi-unmanned aerial vehicles (UAVs) control. Although many studies have introduced several systems for voice-driven UAV control, most have focused on a general speech recognition architecture to control a single UAV. However, for stable voice-controlled driving, it is essential to handle the environmental conditions of UAVs carefully, including environmental noise that deteriorates recognition accuracy, and the operating scheme, e.g., how to direct a target vehicle among multiple UAVs and switch targets using speech commands. To handle these issues, we propose an efficient vehicle-embedded speech recognition front-end for multi-UAV control via voice. First, we propose a noise reduction approach that considers non-stationary noise in outdoor environments. The proposed method improves the conventional minimum mean squared error (MMSE) approach to handle non-stationary noises, e.g., babble and vehicle noises. In addition, we propose a multi-channel voice trigger method that can control multiple UAVs while efficiently directing and switching the target vehicle via speech commands. We evaluated the proposed methods on speech corpora, and the experimental results demonstrate that the proposed methods outperform the conventional approaches. In trigger word detection experiments, our approach yielded approximately 7%, 12%, and 3% relative improvements over spectral subtraction, adaptive comb filtering, and the conventional MMSE, respectively. In addition, the proposed multi-channel voice trigger approach achieved approximately 51% relative improvement over the conventional approach based on a single trigger word.

Keywords: speech recognition; voice-driven control; noise reduction; voice trigger; unmanned aerial vehicle (UAV); multi-UAVs control; minimum mean squared error (MMSE)

1. Introduction

A variety of speech recognition applications have been introduced after the commercial success of personal assistant devices. In particular, considerable attempts have been made to apply voice-driven control for moving vehicles, e.g., cars and airplanes (even combat planes). In addition, the convenience of hands-free voice control has extended the range of speech recognition applications to unmanned aerial vehicles (UAV).

Compared to remote manual control, voice interfaces facilitate rapid and convenient UAV control. Previous studies have proposed several UAV voice-control systems [1–3]. Most conventional studies focused on a general speech recognition architecture to control a single drone using voice commands. However, for stable voice control of UAVs, it is necessary to carefully investigate and handle various environmental conditions.

One of the main environmental issues to be addressed for voice-controlled UAVs is the reduction of background noises captured by microphones. Noise signals that interfere with the speech recognition process may deteriorate recognition accuracy. Thus, most systems handle background noise using noise reduction methods [4,5]. In addition, most conventional methods target a single UAV under the assumption that the speech recognition system is fully assigned to a single UAV. Thus, a new operating scheme is required to direct a target vehicle among multiple UAVs and switch targets without remote control devices when controlling multiple UAVs.

Standard speech recognition systems comprise several modules, including front-end, recognition and post-processing modules (Figure 1) [6,7]. The front-end module requires various fundamental processes, e.g., noise reduction, voice triggering, and acoustic feature extraction. The extracted acoustic features are submitted to a recognition module, where speech recognition is performed using general pattern recognition techniques, e.g., deep neural network (DNN) or hidden Markov model (HMM). The recognition process is followed by the post-processing module, which verifies the recognition result and determines whether the result is accepted or rejected.

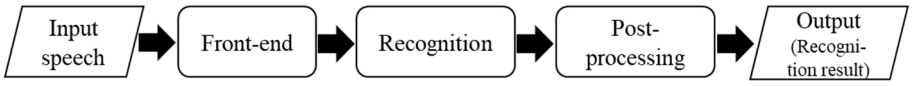


Figure 1. Procedure of standard speech recognition systems.

The two main issues (i.e., the noise reduction and the operation scheme) to be addressed for voice-control of multiple UAVs can be handled by the front-end module. This study proposes an efficient speech recognition front-end for voice-driven multi-UAV control. In particular, we concentrate on a vehicle-embedded recognition scheme rather than a server-based scheme, which can be affected by network capacity and processes an indirect UAV control.

This remainder of this paper is organized as follows. In Section 2, we propose an efficient speech recognition scheme for voice-driven multi-UAV control. Section 3 presents the proposed vehicle-embedded speech recognition front-end. In Section 4, several experiments conducted on speech data and their results are reported and discussed. Finally, Section 5 concludes the paper.

2. Speech Recognition Scheme for Voice-Driven Multi-UAV Control

2.1. Conventional Speech Recognition Schemes

Conventional speech recognition schemes are divided into two types according to the location of the speech recognition engine, i.e., device-embedded and server-centric schemes. In device-embedded speech recognition schemes, all processing modules (from front-end to post-processing) are embedded in user devices. In contrast, the server-centric scheme assigns overall processing to a remote server, e.g., a cloud server. These speech recognition schemes are compared in Figure A1 in Appendix A.

Recent advances in deep learning techniques realized continuous speech recognition at stable performance [8–10]. However, the high computational intensity of deep learning algorithms requires high-performance hardware. Thus, most continuous speech recognition engines operate under the server-centric scheme.

Conventional speech recognition schemes can be applied to voice-driven multi-UAV control, as shown in Figure 2. In the device-embedded scheme, a handheld device (e.g., smartphone, UAV controller) recognizes the user's speech and transfers the command to the UAVs. Thus, a single

device controls all UAVs via speech recognition processes. Although this direct control scheme facilitates rapid control, the device must be able to handle intensive processing burdens.

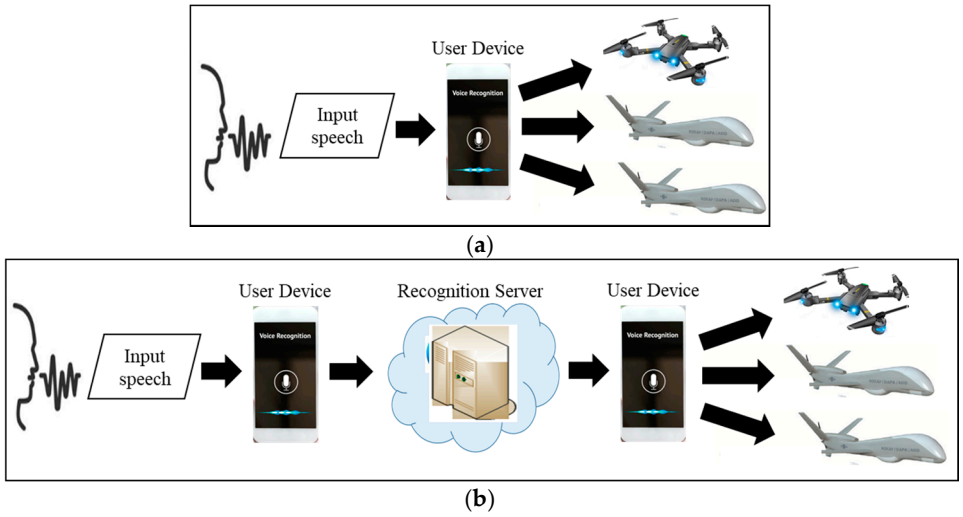


Figure 2. Comparison of (**a**) device-embedded and (**b**) server-centric speech recognition schemes for voice-driven multi-UAV control.

The server-centric scheme depends on the server's hardware capacity. Here, the device transfers the user's input speech to a remote server. Then, the server performs the recognition process and returns the result to the user device. Finally, the device transmits the command to the UAVs. This scheme allows complex command formats with unlimited vocabulary to be processed because the server can handle huge computational loads. However, when multiple users submit speech commands to a single server simultaneously, the server may experience a huge burden, which would delay command transmission. In addition, the server-centric scheme manages indirect UAV control by performing three data transmission sequences: from a user device to the recognition server, from the server to the device, and from the device to UAVs. This indirect communication can incur communication costs, which may induce recognition errors or cause commands to be missed due to packet loss while a user device or UAV moves. Note that the packet loss problem may be more serious for special purpose UAVs (e.g., military UAVs) that move at high speeds.

2.2. Proposed Speech Recognition Scheme for Voice-Driven Multi-UAV Control

To address the drawbacks of conventional speech recognition schemes, this study proposes an efficient recognition scheme for voice-driven multi-UAV control. The proposed scheme is summarized as distributed speech recognition. As shown in Figure 3, the user device and UAVs share speech recognition processes. Here, the user device processes the front-end module, thus producing acoustic features from the input speech. Once the feature is transmitted to the UAVs, a system on the UAVs performs recognition processes following the front-end process.

There are two main reasons the front-end is assigned to the user device. First, the distribution of a sequence of recognition processes mitigates the user device's computational burden that the conventional device-embedded scheme should manage. Next, if the UAV system performs all processes (including front-end processes), the user device should send raw speech data to the UAV. However, the speech data may be degraded due to packet loss problems because speech data are much larger in size compared to feature data.

The mechanism that the UAV system manages recognition processes without passing a recognition server allows the UAV to obtain recognition results directly. Thus, recognition errors and missing commands can be reduced, and the UAV can respond rapidly to user command. In fact, most UAV systems have insufficient computing capacity to handle continuous speech recognition that should

process a number of vocabularies. However, general messages regarding UAV control have a short sentence form comprising several word sequences. Special purpose UAVs have sufficient hardware capacity to process such connected word recognition tasks with a limited number of vocabularies.

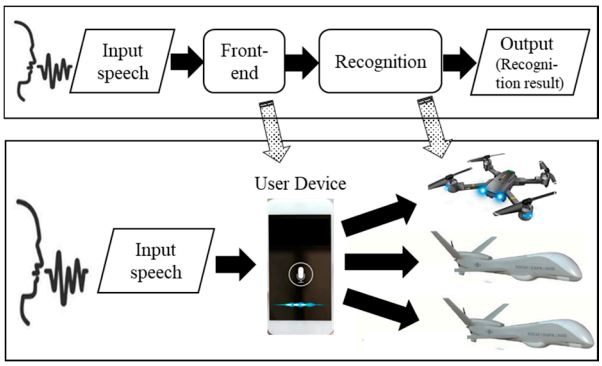


Figure 3. Proposed distributed speech recognition scheme for multi-UAV control.

3. Vehicle-Embedded Speech Recognition Front-End for Multi-UAV Control

In this section, we propose an efficient front-end module that is processed on the user device according to the distributed speech recognition scheme.

3.1. Procedures of Vehicle-Embedded Speech Recognition Front-End

Figure 4 shows the general procedure of the speech recognition front-end. The front-end comprising four main processes: voice activity detection, feature extraction, noise reduction, and voice trigger. The first two processes are necessarily required for speech recognition, whereas noise reduction or voice trigger processes can be skipped according to the system environments and speech recognition purposes.

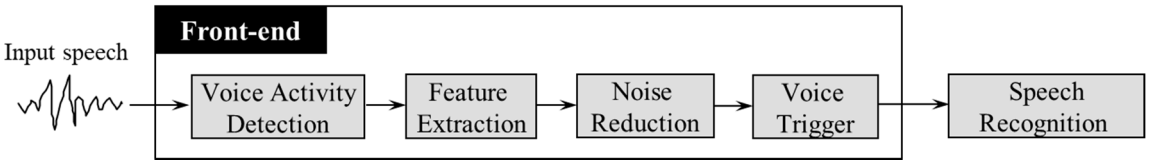


Figure 4. General procedure of speech recognition front-end.

The voice activity detection (VAD) process attempts to detect speech regions to select the target speech data for the speech recognition process. Many studies have investigated VAD [11–17]. Energy-based and zero-crossing rate (ZCR) approaches have been widely used [11–14]. Energy-based VAD considers a general tendency of speech signals whereby the signal energy of speech regions is greater than that of non-speech regions. Thus, this approach detects speech regions by comparing the energy of a given region to a predetermined threshold to differentiate speech and non-speech regions. The ZCR demonstrates a rate of change between positive and negative values in audio signals. This approach considers a general tendency of speech signals where the ZCR of speech regions is greater than that of non-speech regions.

Most VAD approaches introduced in recent years have mainly concentrated on DNN techniques, including CNN, CLDNN, and LSTM-RNN [15–17]. DNN-based VAD approaches are categorized by the types of input features. General approaches use several acoustic features as the DNN input and determine speech or non-speech regions according to DNN procedures. Other approaches use data-driven features, e.g., spectrogram features, and some studies utilize context information of speech data. Although the approaches significantly outperformed simple conventional algorithms, e.g., the energy-based approach, DNN-based approaches incur high computational costs that cannot be processed efficiently on user devices.

The main purpose of a front-end module in a standard speech recognition system is the extraction of acoustic features from the input speech data that represent acoustic characteristics in the time or frequency domains. The features are extracted for a frame, which means a fixed number of speech signals. Thus, the input speech signals are first divided into frame units of equal duration (e.g., 20 or 30 ms), and features are then extracted for each frame. Such features include (but are not limited to) fundamental frequency, energy, and formant frequencies. The most representative acoustic feature for speech recognition is the Mel-frequency cepstral coefficients (MFCC), which describe the spectral characteristics of speech signals.

Feature extraction is followed by the noise reduction and voice trigger processes. In this study, we handle these two processes carefully to achieve reliable performance in multi-UAV environments. The proposed noise reduction and voice trigger approaches are described in Sections 3.2 and 3.3, respectively.

3.2. Noise Reduction for UAV Environments

3.2.1. Conventional Noise Reduction Approaches

The noise reduction process is performed to eliminate noise components in speech regions. Various noise reduction methods have been applied for robust speech recognition in terms of speech enhancement. Such methods can be divided according to the number of microphone channels: single-channel approaches using a single microphone and multi-channel approaches based on a microphone array. Methods that have been studied actively in recent years are based on a multi-channel microphone array to correctly estimate noise signals using relevant techniques, e.g., beamforming and spatial filtering [18–20]. Generally, the microphone array is available for immovable electronic devices in indoor environments, and most are targeted at reducing room reverberation for voice-driven control of smart home devices.

In this study, we primarily target outdoor noises that user devices with a single-channel microphone are exposed to. Two methods have been primarily been applied in single-channel noise reduction, i.e., prior knowledge-based and model-based methods. Each has been widely applied in speech recognition tasks in consideration of operating environments [4]. Prior knowledge-based approaches attempt to estimate noise components and use them to eliminate noises in speech regions. Representative approaches include filtering techniques, e.g., spectral subtraction and adaptive comb filtering, and spectrum reconstruction methods, e.g., the minimum mean squared error (MMSE).

In recent years, many studies have investigated model-based methods. They train mapping properties between clean speech and noise-contaminated speech using a DNN-based regression model and a generative adversarial network (GAN) and then reconstruct the de-noised speech from noisy input speech via DNN decoding [21,22]. Although DNN-based modeling approaches have demonstrated stable performance, they have several limitations relative to our target task.

First, the distributed speech recognition architecture assigns front-end processes (including feature extraction and voice trigger) to the user device. The device also plays a role in reducing background noises from the input speech. Note that computationally intensive DNN decoding may be a difficult process for the user device. Next, the regression model and GAN should maintain noise characteristics to train the mapping properties between clean and noisy speech data. Thus, they provide very stable performance for stationary noises but may perform inefficiently on non-stationary noises captured by the device when the user moves. Finally, if the model-based noise reduction process is assigned to the UAV system rather than the user device, the overall input speech data should be transferred continuously from the device to the UAV. In this case, the data size increases significantly compared to delivering acoustic feature parameters, thereby increasing the frequency of packet loss.

Consequently, this study focuses on prior knowledge-based noise reduction using a single microphone. The most representative method of this technique is spectral subtraction [23,24]. This technique assumes that noise and speech signals are not correlated and combined additively.

Here, if the noise signal characteristics change slowly compared to those of speech signals, the noise components estimated in non-speech regions are used to reduce the noise signals in the speech regions. Let $x(t)$, $s(t)$, and $n(t)$ be noisy speech, clean speech, and additive noise, respectively. The power spectrum of clean speech ($\left|\hat{S}(\omega)\right|$) is estimated by subtracting the spectrum of noise signals ($\left|\hat{N}(\omega)\right|$) from the spectrum of noisy speech ($\left|X(\omega)\right|$) as follows.

$$\left|\hat{S}(\omega)\right| = \begin{cases} \left|X(\omega)\right| - \left|\hat{N}(\omega)\right|, & \text{if } \left|X(\omega)\right| > \left|\hat{N}(\omega)\right| \\ 0, & \textit{otherwise} \end{cases} \quad (1)$$

The adaptive comb filtering-based approach eliminates noise components by enhancing speech harmonics using the fundamental frequency [25]. The filter is based on the observation that speech waveforms are periodic, corresponding closely with the fundamental frequency. While deemphasizing the harmonic valleys of noise-contaminated speech, the filter removes noise components based on the assumption that noise components exist between the harmonics in the speech spectrum. As a result, this filter enhances the spectral magnitude in frequency bins corresponding to the harmonic frequencies. The adaptive comb filter is expressed as follows:

$$h(n) = \sum_{k=-L}^{L} \alpha_k \times \delta(n - N_k) \quad (2)$$

where $\delta(n)$ is a unit sample function and the length of the filter is $2L + 1$. N_k corresponds to the fundamental frequency, and α_k denotes a filter coefficient that satisfies $\sum_{k=-L}^{L} \alpha_k = 1$. Note that the filter coefficient is predetermined by a window function, e.g., the Hamming function.

The filtering approaches are highly dependent on prior knowledge. The spectral subtraction requires correct detection of non-speech regions to estimate noise components in the regions, while adaptive comb filtering depends on the fundamental frequency of the speech signals. Thus, these approaches may be incapable of reducing non-stationary noise or severe noise signals in which detection of non-speech regions or estimation of the fundamental frequency is not available.

The MMSE targets speech components, whereas other approaches concentrate on estimating noise components in non-speech regions [26,27]. This technique estimates noise components in the spectrum and restores clean speech with signal processing level. Thus, it efficiently reduces non-stationary noises.

3.2.2. MMSE Enhancement Based on Spectral Energy Variation for Noise Reduction in UAV Environments

User devices that control UAVs may be exposed to various background noises when users operate UAVs in outdoor environments. Generally, noise components in outdoor environments are non-stationary noises that change rapidly and continuously in the time and frequency domains. In contrast, indoor environments generally produce stationary noises, e.g., the operating sounds of home appliances. Figure A2 in Appendix A compares the two types of noises via a 2-dimensional spectrogram.

Stationary noises can be reduced easily by most conventional noise reduction methods because the noise components estimated in non-speech regions are preserved in speech regions without variation. In contrast, non-stationary noises provide inefficient conditions for noise reduction because it is difficult to classify speech and non-speech regions, and each region has different noise characteristics.

Among conventional approaches, the MMSE method can provide stable noise reduction performance in terms of spectral analysis. It analyzes speech and noise characteristics in each frequency bin of both speech and non-speech regions, whereas other methods consider the overall spectral characteristics in non-speech regions. Thus, the MMSE method is not dependent on classifying speech and non-speech regions. On the other hand, other approaches require detection of non-speech regions to estimate noise components but may fail to detect non-speech regions in speech contaminated by non-stationary noises. Thus, this study concentrates on the MMSE method to handle non-stationary noises in outdoor environments.

Generally, the noise-contaminated speech signal $y(n)$ can be described as $\mathrm{x}(n) + w(n)$, where $\mathrm{x}(n)$ and $w(n)$ denote the clean speech signal and noise signal, respectively. In the frequency domain, it can be transformed to $\left|Y(\omega)\right| = \left|\mathrm{X}(\omega)\right| + \left|\mathrm{W}(\omega)\right|$, where $\left|\mathrm{X}(\omega)\right| = S(\omega)e^{j\varphi(\omega)}$ and $\left|\mathrm{Y}(\omega)\right| = R(\omega)e^{j\varphi(\omega)}$. The main objective of the MMSE method is to estimate de-noised speech $\hat{S}(\omega)$ such that $\left(S(\omega) - \hat{S}(\omega)\right)^2$ is minimized. The MMSE describes $\hat{S}(\omega)$ as follows:

$$\hat{S}(\omega) = G(\omega)R(\omega) \tag{3}$$

where $G(\omega)$ is a gain function used to adjust the noise-contaminated speech to a clean speech signal. If speech components are dominant in the frequency bin ω, $G(\omega)$ approaches 1, thereby maintaining $R(\omega)$. Otherwise, the function proceeds to 0, thereby decreasing $R(\omega)$. Conventional studies introduced several ways to estimate the gain function. However, the following approximation is considered an efficient method in terms of correctness and computational intensity [28,29]:

$$G(\omega) = \frac{\wedge(\omega)}{1 + \wedge(\omega)} \approx 1 - q(\omega) \tag{4}$$

where $\wedge(\omega)$ is a likelihood ratio between speech and non-speech components in frequency bin ω. As described in (4), $G(\omega)$ can be approximated using $q(\omega)$, i.e., the speech absence probability (SAP).

The SAP of the l-th frame in frequency bin ω is estimated as follows:

$$q_l(\omega) = \alpha \cdot q_{l-1}(\omega) + (1 - \alpha) \cdot I_l(\omega) \tag{5}$$

where α is a constant value between 0 and 1, and $I_l(\omega)$ is a hard-decision parameter that determines whether speech is present in the corresponding frequency bin. Here, the decision is made using the posteriori signal-to-noise ratio (SNR) (γ_ω^l) and threshold (γ_{TH}) as follows:

$$I_l(\omega) = \begin{cases} 0 & (\gamma_\omega^l \geq \gamma_{TH}) \\ 1 & (\gamma_\omega^l < \gamma_{TH}) \end{cases}. \tag{6}$$

If $I_l(\omega)$ is determined to be 0, the frequency region is considered a speech-present bin. In this region, $q_l(\omega)$ indicates a small value reaching 0 according to Equation (5), and the gain function approaches 1 according to Equation (4). In contrast, in speech-absent bins, $I_l(\omega)$ becomes 1, thus increasing the SAP but decreasing $G_l(\omega)$.

As described above, the hard-decision parameter $I_l(\omega)$ plays an import role in determining the SAP and gain function, which is a main factor in estimating de-noised speech $\hat{S}(\omega)$ in the MMSE estimator. In other words, an incorrect decision of $I_l(\omega)$ may induce errors in MMSE-based noise reduction. However, the conventional method of determining this parameter has some drawbacks. First, the posteriori SNR (γ_ω^l) used to determine the parameter may be incorrect for non-stationary noises in which spectral characteristics continuously change over time and frequency. The next problem is related to threshold (γ_{TH}), i.e., the conventional method is highly dependent on this threshold, which is maintained as a fixed value. Thus, it is equally applied in every speech frame and frequency bin, and therefore, it may incorrectly operate for non-stationary noises.

To handle the hard-decision parameter carefully, we concentrate on the spectral tendency in terms of energy variation in the time domain. As shown in Figure A2, spectral energy in speech-present frequency bins indicates higher variation than in speech-absent frequency bins in a short period. This tendency is also observed in non-stationary noises because noise components are added to speech-present and speech-absent bins identically during a given period.

To observe the tendency for non-stationary noises, we investigated spectral energy variation for each frequency bin over time, using several frames of speech contaminated by outdoor noises.

The variation was calculated as the variance of spectral energy of each frequency bin for a certain number of consecutive frames. Figure 5 shows the result.

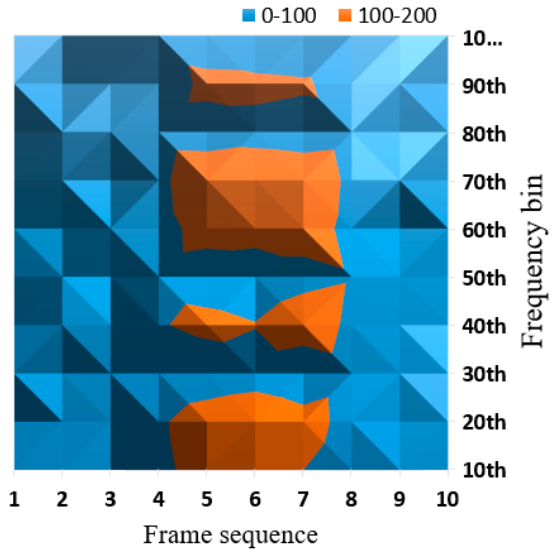


Figure 5. Tendency of spectral energy variation over time.

In this experiment, we attempted to classify each frequency bin into speech-present and speech-absent bins using a criterion value (100). Frequency bins greater than 100 are shown in orange, and other bins are shown in blue. Regions indicating higher values are shown in darker colors. As shown in Figure 5, frame sequences 5 to 7 can be determined as speech regions in which both speech-present and speech-absent bins are observed. Other sequences colored by blue can be regarded as non-speech regions where only noise components are observed. This classification result based on spectral energy variation corresponds approximately to the speech data used in this experiment.

To derive the hard-decision parameter using a mathematical formula, we first define the energy variation of each frequency bin over time. To observe the variation during a given period, we divide input signals as a 20-ms frame unit (Section 3.1) and designate a certain number of consecutive frames as the analysis window. Here, the size of the analysis window is limited to five to ten frames because more frames may fail to detect exact speech regions. In addition, the next window begins from the second frame of the current window, thereby providing overlap between two consecutive windows.

First, the mean spectral energy value is calculated for each frequency bin belonging to a given analysis window as follows:

$$\mu_l(\omega) = \frac{1}{F}\sum_{f=1}^{F}\left|Y_{l,f}(\omega)\right|, \tag{7}$$

where F is the number of frames an analysis window covers, and $\left|Y_{l,f}(\omega)\right|$ is the spectral energy of frequency bin ω of the f-th frame of the l-th analysis window. Then, the spectral energy variance is obtained as follows:

$$\sigma_l{}^2(\omega) = \frac{1}{F}\sum_{f=1}^{F}\left(\left|Y_{l,f}(\omega)\right| - \mu_l(\omega)\right)^2. \tag{8}$$

Finally, we define a new hard-decision parameter to substitute the conventional method described in Equation (6). The proposed parameter uses the spectral energy variance as a measure to determine

if frequency bin ω of the l-th frame is pertinent to a speech-present bin. Here, the standard deviation can replace the variance for down-scaling as follows:

$$I_l(\omega) = \begin{cases} 0 & (\sqrt{{\sigma_l}^2(\omega)} > \sigma_{TH}(\omega)) \\ 1 & (\sqrt{{\sigma_l}^2(\omega)} < \sigma_{TH}(\omega)) \end{cases}, \quad (9)$$

where $\sigma_{TH}(\omega)$ indicates the decision criterion. Each frequency bin has a different criterion to consider the spectral characteristics of speech regions, whereas the criterion of the conventional approach is determined as a fixed value in the overall frequency bins without consideration of the spectral properties. Here, if a frequency bin provides higher standard deviation compared to the threshold estimated in the corresponding bin, $I_l(\omega)$ is determined as 0. Thus, the frequency bin is considered as a speech-present bin. Otherwise, the frequency bin is determined as a speech-absent bin.

In Equation (9), index l can be interpreted differently for $\sqrt{{\sigma_l}^2(\omega)}$ and $I_l(\omega)$, which use l as an index of a given analysis window and an index of a given frame, respectively. In fact, these two indexes move equivalently because the index of a window refers to the index of the first frame of the window, and the following window begins from the second frame of the current window.

The proposed hard-decision parameter based on spectral energy variation is expected to further determine the speech-present and speech-absent bins correctly, thereby improving the correctness of the conventional MMSE method.

3.3. Voice Trigger for Multiple UAVs Control

Generally, a trigger module operates constantly in smart devices with voice interface functions [30]. This module is used to ignore normal conversation speech or background noises while passively listening to sounds. Once it detects a user-spoken trigger word, it sends a message to a remote cloud server to activate a speech recognition engine. The user's spoken data following the word are sent to the server and recognized by the server's recognition engine. The trigger word (also referred to as a wake-word) is predefined.

After recognizing the user's speech, the server formulates an appropriate response and sends the response message to the device. Then, the device forms a response to the user via a synthesized voice. This process allows the user to experience direct communication with the device. A standard voice interface operation using a voice triggering function is provided in Figure A3.

If voice triggering is not provided in voice-driven UAV control systems, users should take physical actions, e.g., pressing a button or touching a display, to activate the speech recognition engine. This triggering process provides users with direct communication with the UAV, thereby increasing convenience and efficiency. In particular, triggering plays a major role in establishing a connection with a target UAV in multi-UAV environments.

3.3.1. Conventional Voice Trigger

The voice trigger is similar to keyword spotting that detects one or more predefined keywords from a sequence of speech signals [31,32]. It is widely used to wake up personal assistant devices. Thus, this task is also known as wake-up-word detection or hotword detection [33,34]. Most conventional voice trigger approaches attempt to detect a single trigger word that is pertinent to the target device [31,33,35]. Some systems can consider multiple trigger words, thus selecting a trigger word and switching to one of other words. However, the device allows a single selected word among multiple trigger words as a wake-up-word because handling a single word is efficient relative to enhancing the detection correctness. Thus, a single acoustic model is sufficient for the operation. In recent years, several studies have attempted to operate multiple trigger words by allowing any predefined multiple words. Here, it is not important to classify which word is detected. Thus, it is sufficient to employ a single model.

Figure 6 illustrates the standard voice trigger procedure when handling a single acoustic model [32]. As can be seen, a single trigger model for a trigger word is used to determine the trigger. For each input voice entered into the device, the trigger module produces a recognition result that represents the similarity between the input voice and the trigger model. The result is compared to a predetermined threshold to determine if the input voice corresponds to the trigger word. When it is determined to be a trigger word, the following voice signals are considered a voice command to recognize. In this study, we refer to this technique as a single-channel voice trigger in terms of the use of a single trigger model.

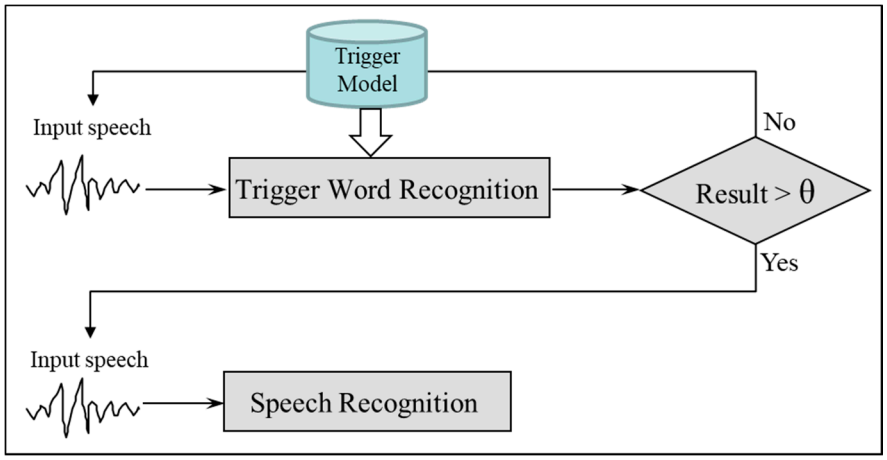


Figure 6. Conventional voice trigger method.

3.3.2. Multi-Channel Voice Trigger for Multi-UAV Control

The conventional voice trigger method is very simple and suitable for operation on user devices; however, it has several limitations in multi-UAV environments. First, the conventional method can only handle a target device using a single trigger word, thereby providing insufficient conditions for multiple UAVs. Next, this approach is highly dependent on a predefined threshold to determine the trigger word. The dependency on this threshold may induce significant determination errors because it may be vulnerable to the speaking variations of multiple users and interference by background noise.

In the distributed speech recognition scheme (Section 2.2), a single user device controls multiple UAVs. Thus, this device should process multiple trigger words to distinguish UAVs. In addition, although general smart devices, e.g., smartphones, tend to be tolerant of trigger recognition errors, such errors may lead to significant damages to UAVs, particularly in military facilities. Therefore, trigger word determination should be processed carefully in UAV environments.

The proposed voice trigger approach for multi-UAV control can be characterized as a multi-channel voice trigger in which each UAV has a distinctive name used as a trigger word, and the user device establishes a connection between the user and the target vehicle among multiple UAVs. The trigger word detection module embedded in the user device has two important purposes, i.e., connecting to a target UAV corresponding to the detected trigger word and waking up the speech recognition engine on the UAV system. Figure A4 illustrates the communication flow of voice trigger based multi-UAV control. Once a trigger word is detected, the device attempts to connect to the corresponding UAV. After receiving an acknowledgment message from the target UAV, the user delivers a speech command. Then, the UAV system recognizes the speech and begins to execute the command. After delivering speech commands, the user can switch to a new target UAV by calling its trigger word.

There are several trigger word detection methods according to the recognition unit. Some systems recognize trigger words based on continuous speech recognition that handles a sub-word unit, e.g., a phoneme or a tri-phone. Such systems require a set of time-consuming tasks, including speech segmentation, sub-word model computation, beam search pruning, and word matching. In this study, we employ word model-based trigger recognition in which trigger word models are

constructed as a recognition unit and only a task of model computation is required. This approach incurs lower computational costs compared to the sub-word model approach, thereby providing sufficient conditions to be processed by the user device.

Figure 7 illustrates the procedures of the proposed multi-channel voice trigger-based front-end and speech recognition for multi-UAV control. The main property of the proposed approach is improving trigger word detection correctness by employing three consecutive processes for filtering non-trigger speech data. Once signals enter the microphone on the user device, the device first detects speech regions using the VAD module. Then, the duration of the detected speech region is compared to predetermined ranges of trigger words using the duration filtering module. If the duration is out of range, the module determines that the given region is pertinent to a non-trigger voice and discards the region. This first filtering process is a soft decision based on a sufficiently large range.

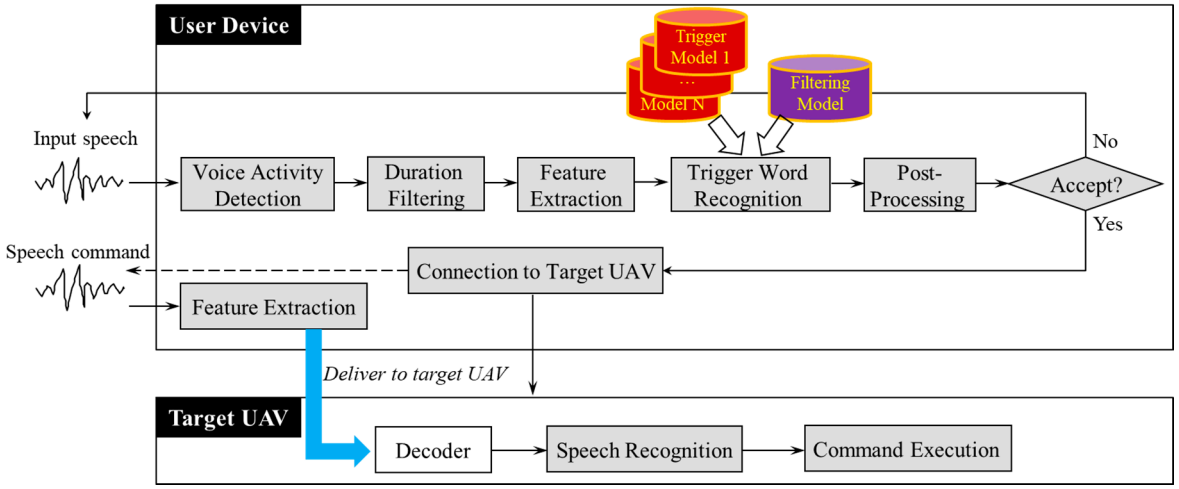


Figure 7. Procedures of proposed multi-channel voice trigger-based front-end and speech recognition.

Once a speech region passes the duration filtering process, acoustic features are extracted from the speech signals and submitted to the following process for trigger word recognition. This second filtering process involves the main feature of the proposed trigger. Here, the trigger recognition module processes two or more trigger word models with the same number as UAVs, whereas the conventional approach only processes a single trigger model (Figure 6). In addition to the trigger word models, we also employ a filtering model that can withdraw non-trigger speech signals.

For acoustic features, each model (i.e., the trigger word models and filtering model) produces a recognition result that represents the similarity between the input speech and each model. Then, the maximum value of the results calculated by the trigger models is compared to the result of the filtering model. If the maximum value is greater than the result of the filtering model, the speech region is determined as the trigger word corresponding to the model giving the maximum value. The other case means that the region demonstrates greater similarity with the filtering model than the trigger word models. Thus, the region is considered to be a non-trigger voice and discarded.

The use of multiple trigger word models enables us to handle two or more trigger words, thereby realizing multi-UAV control. Thus, it can be characterized by a multi-channel voice trigger. The filtering model determines trigger words by filtering out non-trigger voices. This approach allows us to determine trigger words without using a fixed threshold, while the conventional approach is highly dependent on such a threshold.

The abovementioned models are acoustic models used in general speech recognition tasks, and they are constructed prior to the recognition stage, using machine learning techniques, e.g., stochastic modeling or a neural network. Although the DNN-based approach provides stable speech recognition performance, it has high computational complexity. Thus, most conventional DNN approaches have focused on continuous speech recognition of large vocabularies. Our target task is trigger word

recognition with a limited number of trigger words. Thus, a stochastic modeling approach, e.g., HMM, that is capable of rapid recognition with low complexity is optimal.

The final filtering process is performed by the post-processing module, which attempts to verify the correctness of the detected trigger word. Thus, we designate the trigger voice region determined in the previous recognition process as a candidate trigger region. A main role of the final process is to determine if the candidate region is finally accepted as a trigger word or rejected as a recognition error. For the decision criterion, we employ the likelihood results produced as an HMM-based speech recognition result. The likelihood results refer to the observation probability, which indicates how closely the speech signals are observed in a given model. After obtaining the observation probability for each model, the results are ranked according to the values. The recognition result is determined as a model with the highest probability, i.e., the top rank in the likelihood results.

Figure 8 illustrates the overall procedures of the proposed post-processing for trigger word verification. The verification is performed using a decision criterion based on the likelihood results. The following equations describe the proposed decision criteria.

$$DC_1(x) = \frac{P(x|R_1(x))}{\sum_{r=1}^{N} P(x|R_r(x))}, \tag{10}$$

$$DC_2(x) = P(x|R_1(x)) - \frac{1}{N}\sum_{r=1}^{N} P(x|R_r(x)), \tag{11}$$

where $R_r(x)$ denotes a model at the r-th rank in the likelihood results estimated from the trigger models and a filtering model for candidate trigger region x. $P(x|R_r(x))$ denotes the observation probability estimated for $R_r(x)$. The two criterion functions calculate the difference between the observation probability in the top-ranked model ($R_1(x)$) and those in other models. This idea considers a general property of observation probability obtained in recognition procedure. Generally, more confidently recognized speech indicates a higher probability in the top-ranked model, thereby making a large difference from probability in other models. The first criterion (10) observes the ratio between two probabilities, and the second criterion (11) considers the direct difference between these probabilities.

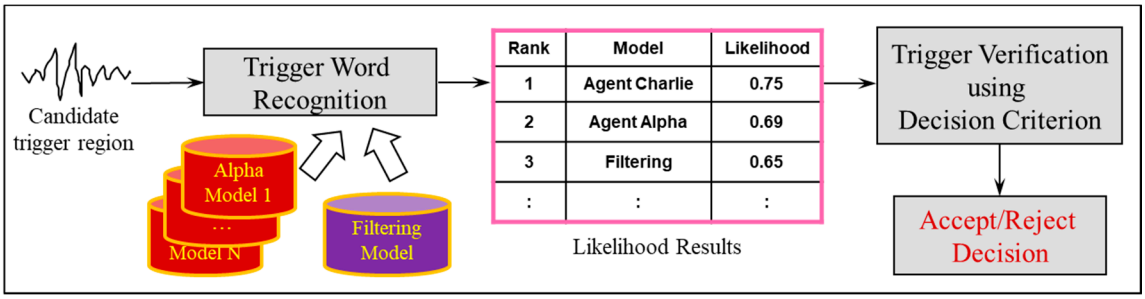


Figure 8. Procedure of proposed post-processing for trigger word verification.

In the trigger verification process, the value estimated by the decision criterion is compared to a decision threshold that is determined empirically. If the value is greater than the threshold, the candidate trigger region is accepted and considered a discriminative trigger word. Otherwise, the region is rejected.

The proposed consecutive filtering processes are expected to improve confidence in detecting trigger word regions and disregarding non-trigger speech regions. Once a trigger word is finally detected, the next procedures are performed, as shown in Figure 7. The user device attempts to connect to a target UAV corresponding to the trigger word. After connecting to the target UAV, the user speaks a control command. The device extracts acoustic features from the speech command, and then delivers

the features encoded in packet data to the target UAV. After receiving the feature data, the UAV system initiates the speech recognition process and executes the command.

4. Experiments and Discussion

To validate the efficiency of the proposed approaches, we conducted several noise reduction and trigger recognition experiments.

4.1. Validation of Proposed Noise Reduction Approach

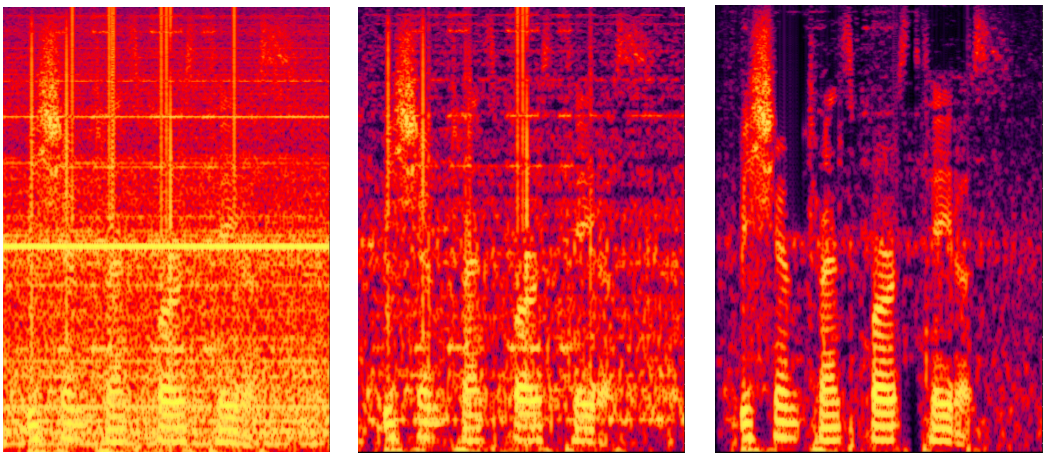
4.1.1. Experimental Setup

To investigate the performance of the proposed noise reduction approach, we used non-stationary noise data obtained from two noise databases that are widely used in robust speech recognition tasks: NOISEX-92 and AURORA [36,37]. NOISEX-92, which was produced by the NATO research study group on speech processing, comprises a set of recordings of eight different noises, e.g., babble, factory noise, and F16 fighter jet noise. AURORA, which was developed by the ETSI, comprises more types of background noises, e.g., subway, car, and airport noise. Among the noise types, we selected five non-stationary noises: babble, F16 fighter jet, subway, airport, and street.

We simulated noise-contaminated speech by adding the noise signals to speech data recorded in clean environments. To consider the noise levels, we varied the intensity of the noises, thereby producing noise-contaminated speech with four SNR conditions: 0 dB, 5 dB, 10 dB, and 15 dB. The 0 dB SNR demonstrates a very severe noise condition, and the 15-dB speech is close to clean speech with little noise.

4.1.2. Experimental Results and Discussion

We compared the noise reduction performance of the proposed approach to that of several conventional approaches, including spectral subtraction, adaptive comb filtering, and MMSE. First, we observed the spectrogram of the speech contaminated by the F16 fighter jet engine noise and de-noised speech by spectral subtraction and the proposed method. Figure 9 shows the result. As can be seen, two noise reduction approaches successfully eliminated the noise components. The proposed method demonstrated notable noise reduction performance in the overall time and frequency regions. In contrast, spectral subtraction did not perfectly reduce the noise components in the speech regions.



(**a**) (**b**) (**c**)

Figure 9. Comparison of spectral figures according to noise reduction approaches: (**a**) original noisy speech; (**b**) de-noised speech by spectral subtraction; (**c**) de-noised speech by proposed approach.

Several mathematical measurements, e.g., spectral distance (SD) and SNR, provide an efficient way to compare noise reduction performance. The SD calculates the distance between clean and de-noised speech in terms of spectral energy as follows:

$$SD = \sqrt{\frac{1}{F}\sum_{\omega=1}^{F}\left[10\log_{10}\frac{X(\omega)^2}{\hat{X}(\omega)^2}\right]^2}, \tag{12}$$

where $X(\omega)$ and $\hat{X}(\omega)$ are the spectral energy of clean and de-noised speech in frequency bin ω, respectively. This measure calculates the SD for the overall spectral energy of F frequency bins for a given frame. If the noise components are eliminated perfectly, the distance between $X(\omega)$ and $\hat{X}(\omega)$ is very small. Thus, the SD value approaches 0.

Figure 10 shows the noise reduction performance of each approach in terms of SD. As can be seen, the proposed MMSE approach achieved outstanding performance compared to the conventional approaches including the conventional MMSE. The proposed approach outperformed the conventional MMSE notably at severe noise levels (low SNR values) than higher SNR values, which indicates that the proposed hard-decision parameter correctly determines speech-present and speech-absent bins in severe noise environments. The conventional MMSE outperformed spectral subtraction and adaptive comb filtering, which demonstrates that the MMSE technique is very efficient at reducing non-stationary noises. The two approaches demonstrated different performances according to the SNR conditions. At low SNR values, spectral subtraction outperformed adaptive comb filtering, while an opposite tendency was observed at high SNR values. This result can be analyzed according to the property of adaptive comb filtering, which is highly dependent on the fundamental frequency extracted from the speech region. It is difficult to estimate the fundamental frequency correctly in severe noisy speech. Thus, filtering may induce adverse noise reduction under low SNR conditions.

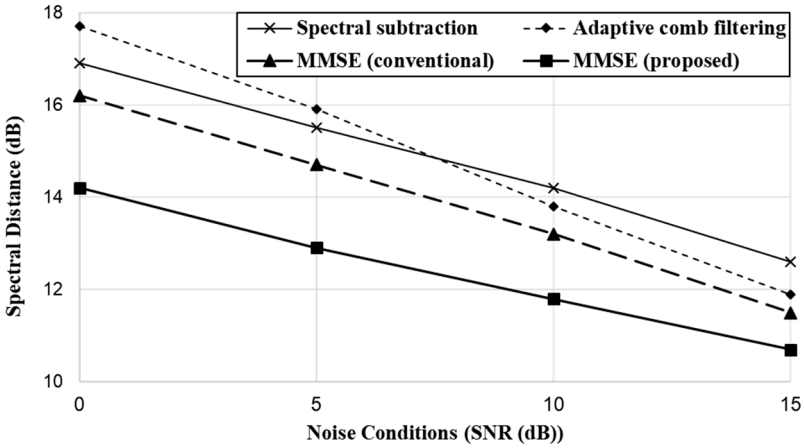


Figure 10. Comparison of noise reduction performance based on SD.

Next, we investigated noise reduction performance based on SNR. The SNR is a ratio of speech signals to noise signals contained in noisy speech signals. Thus, high SNR values are observed when fewer noise components are present. The SNR can be calculated directly from the input signals as follows:

$$SNR = 10\log_{10}\frac{\sum_{n=1}^{N} x(n)^2}{\sum_{n=1}^{N}\left(\hat{x}(n)-x(n)\right)^2}, \tag{13}$$

where $x(n)$ and $\hat{x}(n)$ refer to clean speech and de-noised speech signals, respectively. This measurement calculates the ratio of clean speech signals to residual noise signals after noise reduction. If the noise signals are reduced perfectly, the difference between $\hat{x}(n)$ and $x(n)$ approaches 0, thereby increasing the SNR value.

Figure 11 shows the noise reduction performance of each method in terms of SNR. As can be seen, performance demonstrated a similar tendency to the SD results. The two MMSE approaches outperformed the other approaches. The proposed MMSE yielded the best performance. As observed in these two results, the proposed approach achieved a significant performance improvement compared to the conventional MMSE in overall noise conditions, and the results support our expectation that the proposed hard-decision parameter based on spectral energy variation more correctly determines speech-present and speech-absent bins than the posterior SNR-based decision approach. Finally, we investigated the performance of the proposed approach according to the size of the analysis window. The result and discussion for this experiment is described in Appendix B.

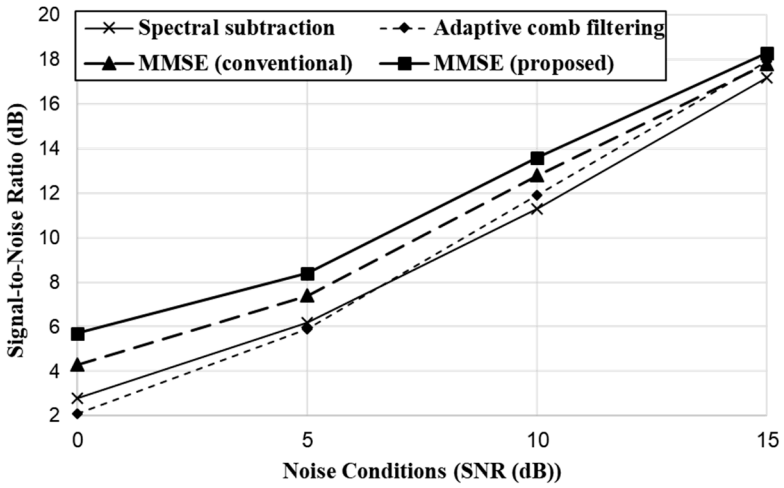


Figure 11. Comparison of noise reduction performance based on SNR.

We also examined noise reduction performance in the context of speech recognition task. With the proposed speech recognition architecture, the user device performs the voice trigger process. Thus, we investigated the performance improvement of voice trigger detection for noisy speech for each noise reduction method. In this experiment, we simulated noisy speech data by adding different types of noise data to clean speech recorded in a silent environment while adjusting noise conditions (0 dB to 15 dB in SNR). Here, the speech data comprised general conversation speech of five participants recorded for approximately two hours, including 500 utterances corresponding to a predefined trigger word ("alpha"). The experimental setup for trigger word detection (including the acoustic models) is described in Section 4.2.

Figure 12 shows the trigger word detection performance in terms of equal error rate (EER). The EER of the original noisy speech ("No processing") is compared to the performance of de-noised speech processed by the conventional and proposed noise reduction methods. As can be seen, noise reduction processing significantly improved trigger word detection accuracy for overall noise conditions. Note that more improvement was observed under severe noise conditions. Among noise reduction methods, the proposed MMSE method outperformed the other method, yielding approximately 7%, 12%, and 3% relative improvement over spectral subtraction, adaptive comb filtering and the conventional MMSE methods, respectively. We found that adaptive filtering and spectral subtraction provided poor accuracy, which indicates that spectral filtering approaches may induce signal distortions in speech while reducing noise components. This tendency was particularly notable in severely noisy speech, which was also observed in the previous results based on spectral measures.

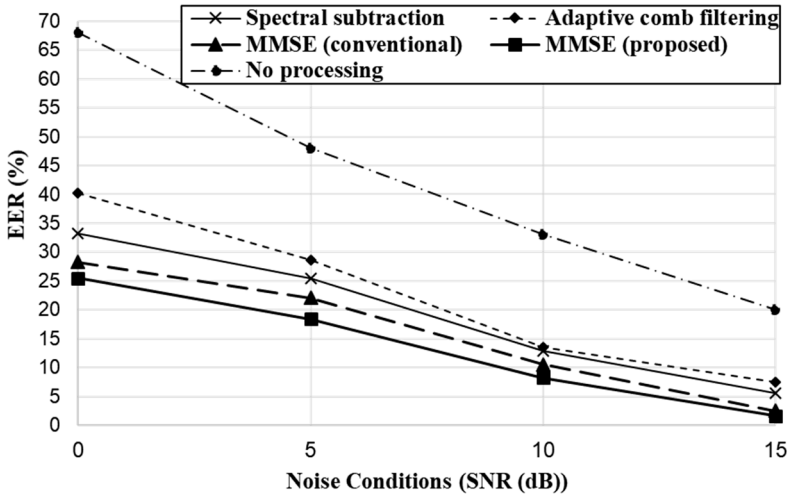


Figure 12. Trigger word detection performance with noisy speech.

4.2. Validation of Proposed Voice Trigger Approach

4.2.1. Experimental Setup

We performed trigger word detection experiments to validate the efficiency of the proposed voice trigger approach. To construct acoustic models for trigger words and a filtering model, training data were collected by 50 persons who uttered predefined trigger words three times and general conversation speech. Here, we selected five trigger words to control five UAVs. The trigger words were defined as standard code words: "alpha," "bravo," "Charlie," "delta," and "echo."

The trigger models and the filtering model were trained with HMM using 750 utterances and approximately 2500 utterances, respectively. Here, each trigger word model was trained with 150 utterances. As a result, we constructed five HMMs for trigger words and an HMM for filtering according to the standard training procedure based on the hidden Markov model toolkit. The optimal numbers of states and Gaussian mixtures were determined during the training phase.

The acoustic feature parameters were configured as a 39-dimensional vector comprising 12-dimensional MFCCs, log-energy, and their first and second derivatives. With regard to the VAD process, we employed a spectral energy-based approach in consideration of the computational capacity of the user device (Section 3.1). The threshold for VAD was determined empirically to avoid the effect of VAD in the verification of the proposed approaches.

For the recognition test, we set up experimental environments to simulate a user device connecting to five remote UAV systems designated with five trigger words, respectively (Figure A5). The test was conducted in a real-time and online manner by 20 participants who did not participate in recording the training data. Each participant spoke general speech regardless of the trigger words to a device and intermittently spoke one of the five trigger words. To observe the performance in adverse environments, noise sounds were played while the participants made conversation. The SNR of the noisy speech was ranged from approximately 5 dB to 10 dB.

Here, when the device detects a trigger word, a system corresponding to the trigger word displays a connection result. A situation of establishing a connection between the device and UAV systems after trigger word detection is shown in Figure A6. In this test, we counted the frequency of correct detection and investigated the missing rate.

4.2.2. Results and Discussion

The performance of the standard speech recognition system is evaluated relative to recognition accuracy. The evaluation of trigger word recognition differs from speech recognition because this task does not require recognizing all input speech. The trigger recognition system has two goals, i.e.,

correctly detecting and recognizing trigger words, and correctly disregarding non-trigger utterances. Thus, general trigger recognition tasks use two errors for performance evaluation. The first error type occurs when the system detects a non-trigger utterance as a trigger word, and the second error type occurs when the system disregards a trigger word region as a non-trigger utterance. Such errors are referred to as false alarm and false rejection errors, respectively.

In consideration of the above error types, trigger word detection performance is evaluated using the detection error tradeoff (DET) curve. The DET curve is used to observe the tendency of the two error types. Generally, the false rejection error rate is inversely proportional to the false alarm error rate because these two errors occur in an opposite manner.

We evaluated the performance of three comparative approaches, i.e., the conventional approach, the proposed approach based on the first decision criterion (Proposed_DC1) described in (10), and the proposed approach based on the second decision criterion (Proposed_DC2) described in (11). The conventional voice trigger approach recognizes only a trigger word, depending on the trigger model, and uses a threshold to determine whether the recognition result is accepted or rejected (Figure 6). We consider this provides more advantages to the conventional task than handling five trigger word models simultaneously because a single model provides better conditions relative to detection performance compared to multiple models. In this experiment, we prepared five programs, each of which handled one of five trigger words. Thus, the programs were switched five times according to the trigger words such that a given program having a trigger word model could detect trigger words that are relevant to the given model. This was expected to provide better detection performance compared to when all five models participated simultaneously. That can be considered an upper bound in detection performance of the conventional approach. Note that the average of five different detection results was determined as the performance of the conventional approach.

The proposed approaches attempt to recognize multiple trigger words with multiple models and a filtering model (Figure 7). In addition, the proposed approaches perform three consecutive filtering processes, including the verification process for all candidate trigger regions in post-processing (Figure 8). The programs for the proposed approaches are maintained to operate five trigger word models simultaneously for all experiments with the aim of detection and identification of trigger words.

Figure 13 shows the DET curve for three approaches. A curve approaching the origin in the coordinates demonstrates better performance, thereby representing lower error rates. As can be seen, the proposed approaches significantly outperformed the conventional approach relative to both false alarm and false rejection errors. The two decision criteria were observed to function differently in trigger word detection. The second criterion provided better performance compared to the first, which represents a direct difference between the observation probability at the first rank in the likelihood results, and the other probabilities provide a better decision criterion than the ratio between them.

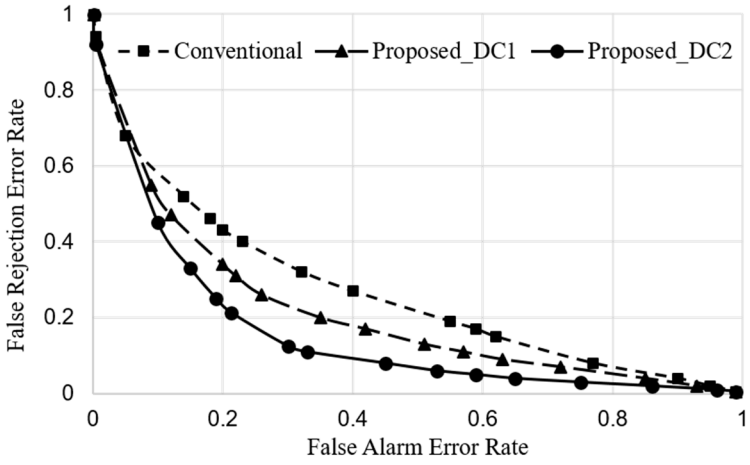


Figure 13. Trigger word detection performance in terms of DET curve.

Next, we investigated EER as another evaluation method. The EER represents an error rate in which the two types of errors indicate the same value. In contrast to the DET curve, which observes the overall performance tendency of the two errors, the EER provides a more intuitive measure of performance. Table 1 summarizes the performance of three different approaches in terms of EER. As confirmed in the DET results, the conventional approach demonstrated higher EER compared to the proposed approaches. The second decision criterion outperformed the first criterion. The proposed approach with the first decision criterion demonstrated approximately 23% relative improvement over the conventional approach, and the approach based on the second criterion achieved approximately 51% relative improvement. In addition, the second criterion provided approximately 22% relative improvement over the first criterion.

Table 1. EER (%) of trigger word detection.

Approach	Equal Error Rate
Conventional	32.1%
Proposed_DC1	26.0%
Proposed_DC2	21.3%

This experimental performance was obtained from the results of the test data for all trigger words. Note that it is necessary to observe trigger word detection performance in a sophisticated manner for each trigger word. For this experiment, we counted the number of correct detections of each trigger word and compared that to the actual number of trigger words in the test data.

Figure 14 shows the results for the five trigger words. The same number of each trigger word was included in the evaluation dataset. The proposed approach with the second decision criterion successfully detected trigger words at approximately 79% accuracy, while the conventional and the first criterion approaches obtained approximately 68% and 74% detection accuracy, respectively. The proposed approaches outperformed the conventional approach in overall trigger words.

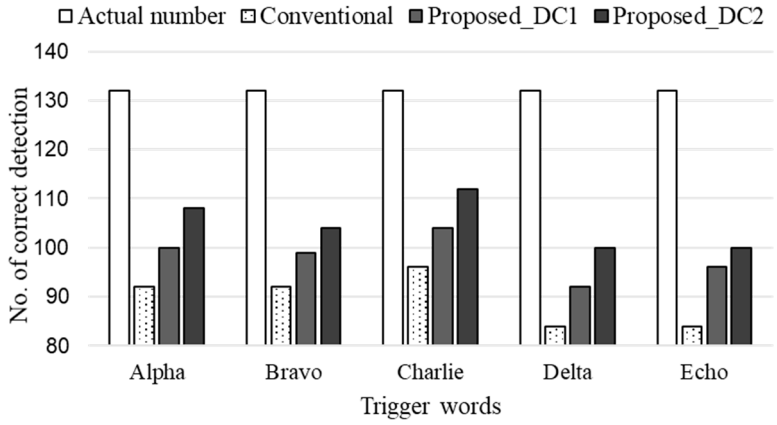


Figure 14. Detection accuracy for respective trigger words.

The five trigger words demonstrated different trigger word detection performance. For example, "alpha," bravo," and "Charlie" showed better performance than the other two trigger words. The conventional approach detected only 84 of 132 utterances of "delta" and "echo," providing approximately 64% detection accuracy. In contrast, the proposed approaches provided greater accuracy than 70% for all trigger words. This result indicates that the proposed approaches provide reliable trigger word detection performance for the target trigger words, thereby facilitating a multi-channel voice trigger for speech-driven control of multiple UAVs.

Next, we compared the performance to that of a DNN-based approach. Here, we trained a standard CNN that performs detection and identification of trigger words while filtering silence and

non-keyword utterances. To facilitate a fair evaluation, this experiment was conducted on clean speech data recorded by five participants in a silent environment. Each participant uttered general conversation speech for approximately two hours while randomly uttering each trigger word 100 times.

The layer configuration for this operation is shown in Figure A7. Here, a bundle of 13-dimensional MFCCs extracted from the input speech enters the input layer. The data pass through three convolutional and max pooling layers sequentially. Then, the dimension of the weighted values is reduced by a pooling layer using the max pooling approach. Dropout is then performed to exclude some data to reduce overfitting. While selecting hyperparameters, we empirically set the optimal filter size to 2×2 and changed the number of filters from 64 to 256 while maintaining a dropout rate of 0.5. The outputs of the convolutional layers enter a fully connected layer. Here, three fully connected layers are followed by a final softmax layer in which the input speech data are categorized as one of the five trigger words or disregarded as a non-trigger utterance. While learning the CNN-based models, we empirically set other hyperparameters as follows: batch size of input features, 800; learning rate, 0.001; and number of epochs, 500. The entire learning process was executed using the Pytorch framework [38].

Table 2 presents the trigger word detection results of five participants in terms of EER. All approaches demonstrated much better performance compared to the results shown in Table 1 because they were evaluated on clean speech data in this experiment. The CNN-based method slightly outperformed the proposed approach for most participants, providing approximately 1% relative improvement. Next, we investigated the detection accuracy for each trigger word. As shown in Figure 15, the CNN method provided better accuracy compared to the MMSE-based approach. However, the performance gap differed according to the trigger words. For the "Charlie" trigger word, the proposed approach missed only three utterances compared to the CNN-based approach. As confirmed in Figure 14, this result demonstrates different performance according to trigger words, e.g., "Charlie" and "alpha" provided better accuracy than the other trigger words.

Table 2. EER (%) of trigger word detection.

Approach	Equal Error Rate (%)					
	Prticipant1	**Participant2**	**Participant3**	**Participant4**	**Participant5**	**Average**
CNN-based	4.32	3.76	4.93	7.29	4.68	4.99
Proposed	5.78	5.12	4.84	8.61	5.43	5.96

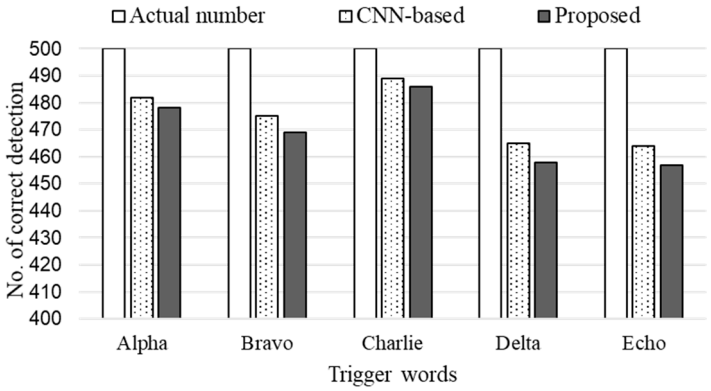


Figure 15. Detection accuracy of each trigger word.

The experimental results for the two approaches indicate that the CNN-based approach provides relatively reliable performance; however, the difference is not significant. Although more sophisticated approaches employing DNNs have been reported to improve the correctness of trigger word detection, such computationally intensive algorithms would be highly inefficient on a user device that continuously monitors incoming speech signals and determines if the signals are trigger words. To consider the difference between the two approaches in terms of computational complexity, we attempted to analyze

the number of parameters required in the training phase and the quantity of computation required during the decoding phase. Table 3 summarizes the results.

Table 3. Computation complexity of trigger word detection approaches.

	Number of Parameters (Training Phase)	Computational Intensity (Decoding Phase)
CNN-based	199,936	513,536
Proposed	120,192	9984

CNN-based approaches require a number of parameters and have high computational intensity. As shown in Figure A7, the CNN-based detection processes seven layers (i.e., three layers for feature extraction, three layers for classification, and one layer for softmax). The overall number of parameters to be estimated was calculated as 199,936, including 1792 parameters in the feature extraction layers ($64 \times 4 + 128 \times 4 + 256 \times 4$), 196,608 parameters in the classification layers ($256 \times 256 \times 3$), and 1536 parameters in the softmax layer (256×6). Note that the number of parameters increases with increasing numbers of convolution filters and layers. In terms of computational intensity during the decoding phase, the CNN-based model required greater than 500,000 computations, which were obtained by considering filter size, the number of input channels, and the input size. The feature extraction layers required 315,392 computations ($64 \times 13 \times 64 + 128 \times 64 \times 16 + 256 \times 128 \times 4$), and the classification and softmax layers performed 198,144 computations ($256 \times 256 \times 3 + 256 \times 6$).

In the proposed approach with statistic modeling (e.g., the HMM), the parameters to be estimated during the training phase comprise the mean and variance of each GMM in each state and a transition probability matrix. When trained with a sufficient number of HMM parameters, eight states and 32 GMMs could handle utterances passing through the duration filtering process. In that case, the total number of parameters to be estimated for five trigger word models and a filtering model was 120,192, comprising 119,808 state parameters ($8 \times 32 \times 78 \times 6$) and 384 transition matrix parameters ($8 \times 8 \times 6$). In terms of computational intensity during the decoding phase, only 9984 computations ($8 \times 32 \times 39$) were required, which represents a significant reduction of computational complexity compared to the CNN-based approach.

4.3. Discussion on Evaluation Results

The main purpose of this paper is to improve the performance of the speech recognition front-end for voice-driven control of multiple UAVs. In particular, we concentrated on two fundamental processes of the front-end, i.e., to reduce background noises and to direct a target vehicle among multiple UAVs and switch targets using speech commands.

In experimental results for the proposed noise reduction approach, our approach showed notable noise reduction performance compared to several conventional approaches, as addressed in Section 4.1.2. In evaluation based on SD, the proposed approach achieved 19%, 25%, and 14% relative improvement over spectral subtraction, adaptive comb filtering and the conventional MMSE, respectively, in 0 dB SNR condition. In SNR-based evaluation, the approach provided further significant performance improvement, achieving 51%, 63%, and 25% relative improvement over each of three conventional approaches in 0 dB SNR condition.

Next, with regard to the proposed multi-channel voice trigger approach, we evaluated how well the approach detects five trigger words pertinent to five target UAVs, as addressed in Section 4.2.2. Our approach outperformed the conventional approach based on a single trigger word, achieving 51% relative improvement.

Finally, we investigated the trigger word detection performance for noisy speech to verify the efficiency of the proposed noise reduction along with the trigger detection approach. As demonstrated in Figure 12, the proposed noise reduction approach showed superior trigger detection accuracy

compared to the other approaches, yielding 7%, 12%, and 3% relative improvement over spectral subtraction, adaptive comb filtering, and the conventional MMSE, respectively.

We attempted to compare the performance of the proposed approach to conventional studies. However, there were few studies on trigger word detection handling multiple trigger words, as addressed in Section 3.3. Most studies have focused on the detection of a single trigger word. In addition, we also address noise reduction for trigger word detection in this study.

A recent study proposed a noise cancellation method for robust trigger detection [34]. This study used a microphone array (multi-microphone) for noise reduction and handled a single trigger word. In contrast, our proposed approach targets more difficult environments in which we attempted to conduct a single microphone-based noise reduction and detect multiple trigger words simultaneously. Nevertheless, we compared the performance of our approach to that in [34]. Among several experiments that we conducted, the trigger word detection performance with noisy speech (Figure 12) could be used because this experiment was performed to investigate the trigger detection performance for a single trigger word ("alpha") for noisy speech with the proposed noise reduction method.

The conventional study [34] reported that the EER ranged from 10% to 20% according to microphone setting for TV noise dataset. As shown in Figure 12, our approach demonstrated similar performance when noise conditions are 5 dB and 10 dB in terms of SNR. This result indicates that the proposed approach demonstrated reliable performance, even though we used further severe noise data recorded in outdoor environments. In particular, our approach requires less amount of computational intensity because the approach employs a simple stochastic model for trigger word detection.

5. Conclusions

In this paper, we have proposed several efficient approaches for voice-driven control of multiple UAVs. First, we proposed an efficient speech recognition scheme based on distributed speech recognition. In this scheme, the user device and UAV system share the recognition processes. Here, the user device processes the front-end module and the UAV system processes the recognition module. In this study, we focused on the front-end module, in which noise reduction and voice trigger functionalities should be carefully handled in consideration of environmental conditions during multi-UAV control.

To handle non-stationary noises in outdoor environments, we enhanced the conventional MMSE approach in consideration of spectral energy variation. To overcome the drawbacks of the conventional hard-decision parameter in the MMSE technique, we employed the spectral energy variance in each frequency bin to determine speech-present and speech-absent bins. In addition, we proposed a new voice trigger approach in which multiple acoustic models are processed to handle multiple UAVs and three consecutive filtering processes are performed to detect trigger words and disregard non-trigger words.

To verify the proposed approach, we developed a simulation environment in which a user device controlled multiple UAV systems via a wireless network, and we performed several experiments on speech and noise data. We found that the proposed approaches outperformed the conventional approaches. The proposed noise reduction approach outperformed conventional approaches, including spectral subtraction and adaptive comb filtering, thereby demonstrating reliable performance in a reduction of non-stationary noises, e.g., babble and F16 fighter jet noises. In addition, the proposed voice trigger approach outperformed the conventional approach based on a single word trigger. The proposed multi-channel voice trigger approach provided reliable trigger word detection accuracy for the target trigger words.

In future work, we plan to investigate efficient approaches for the speech recognition process performed on UAV systems.

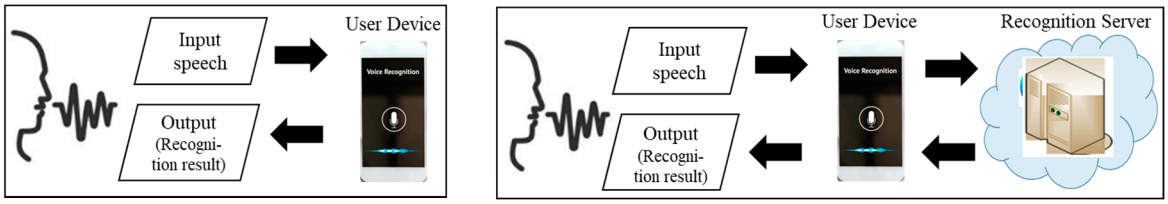
Author Contributions: Conceptualization, J.-S.P.; methodology, J.-S.P.; software, J.-S.P. and H.-J.N.; validation, J.-S.P. and H.-J.N.; formal analysis, J.-S.P.; writing—original draft preparation, J.-S.P.; writing—review and editing,

J.-S.P.; supervision, J.-S.P.; project administration, J.-S.P.; funding acquisition, J.-S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Hankuk University of Foreign Studies Research Fund, the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1A2C1013162), the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00313) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

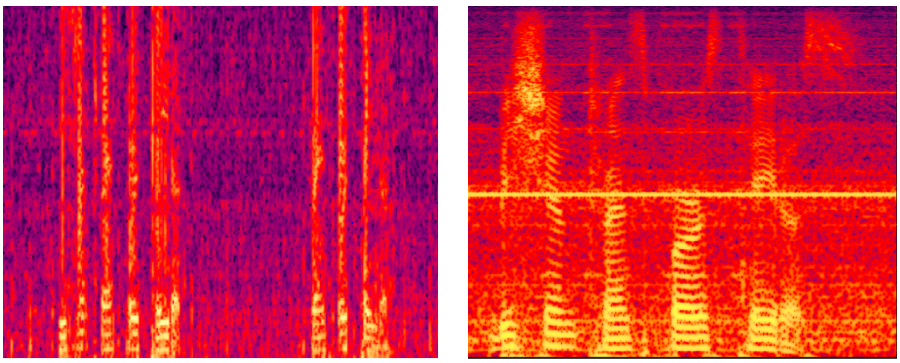
Conflicts of Interest: The authors declare no conflict of interest.

Appendix A



(**a**) Device-embedded (**b**) Server-centric

Figure A1. Comparison of device-embedded and server-centric speech recognition schemes [6].



(**a**) Stationary noise (**b**) Non-stationary noise

Figure A2. Comparison of speech data contaminated by (**a**) stationary noise in indoor environments and (**b**) non-stationary noise in outdoor environments. These figures represent a two-dimensional spectrogram that describes the spectral energy tendency of speech and non-speech signals with color according to the time (horizontal axis) and frequency (vertical axis) domains.

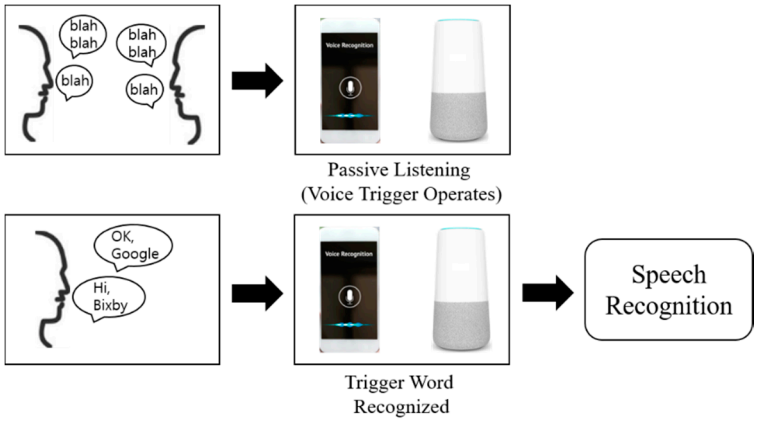


Figure A3. Standard voice interface operation in smart devices [30].

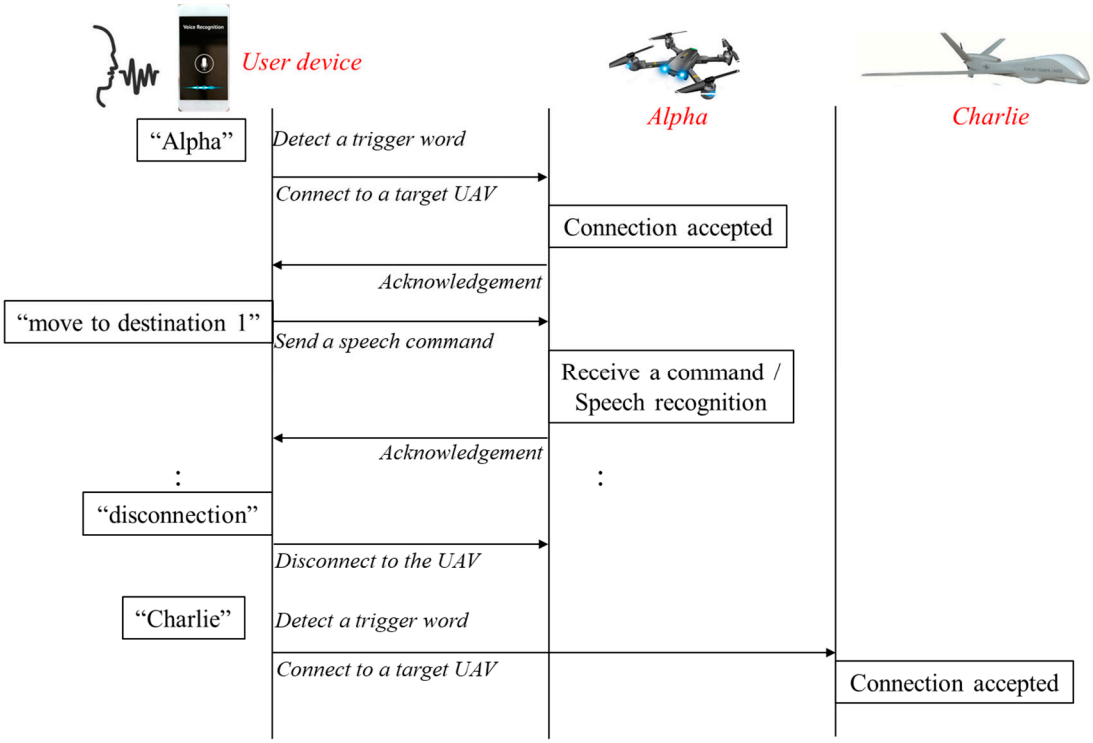


Figure A4. Communication flow of voice trigger-based multi-UAV control.

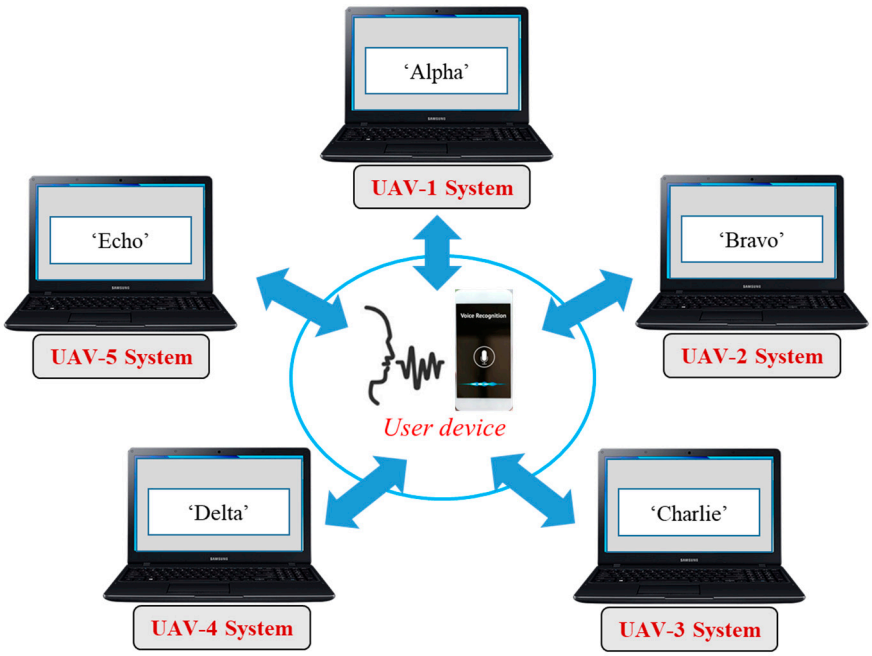


Figure A5. Experimental environments simulating user device and five UAV systems.

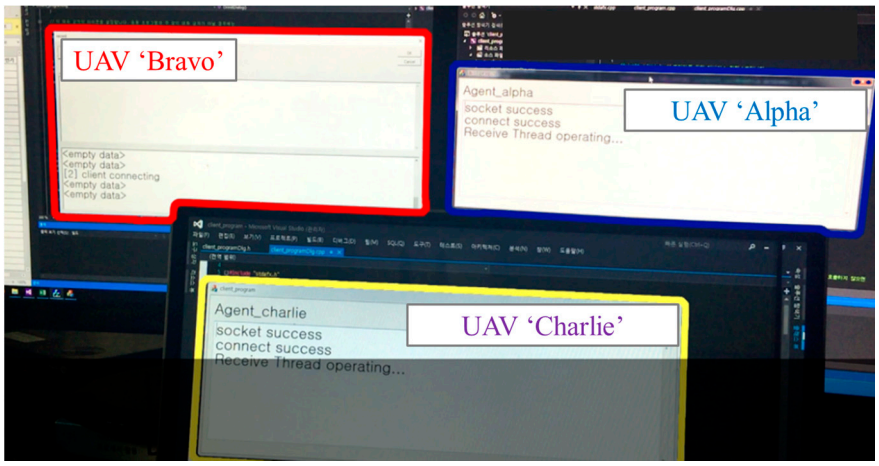


Figure A6. Establishing connection to UAV systems.

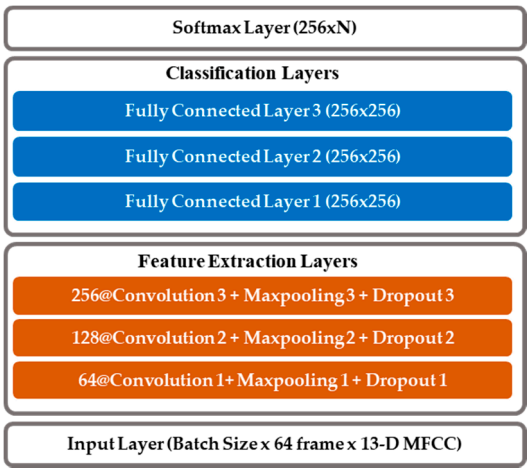


Figure A7. Layer configuration for CNN-based trigger word detection.

Appendix B

With regard to the performance evaluation, the performance of our proposed noise reduction approach according to the size of the analysis window was investigated. As discussed in Section 3.2.2, the frames in an analysis window participate in estimating spectral energy variation. We are confident the duration of the window, that means the number of frames, affects the correctness of the hard-decision of speech-present and absent bins. Here, we observed the SD in the same manner as Figure 10 while varying the number of frames in the analysis window from five to 11. Figure A8 shows the results. The proposed approach demonstrated significant differences in noise reduction performance according to the number of frames in the analysis window. The result of five frames showed the worst performance. Thus, five frames is not long enough to provide sufficient information about spectral energy variation to determine speech-present and absent bins. In contrast, results obtained with longer durations provided better performance. In particular, higher SNR values yielded significant performance differences among duration sets, and the results were similar with low SNR values. This indicates that the duration does not affect performance significantly under severe noise conditions. The best performance was observed with nine frames. However, the result obtained with 11 frames was degraded. Thus, we consider nine frames to be the optimal analysis window size.

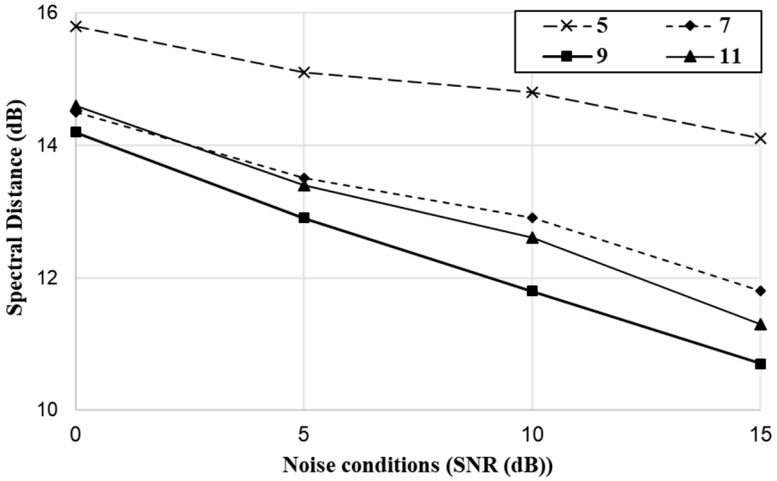


Figure A8. Performance of proposed MMSE approach according to analysis window duration.

References

1. Oneata, D.; Cucu, H. Kite: Automatic speech recognition for unmanned aerial vehicles. *arXiv* **2019**, arXiv:1907.01195.
2. Contreras, R.; Ayala, A.; Cruz, F. Unmanned aerial vehicle control through domain-based automatic speech recognition. *Computers* **2020**, *9*, 75. [CrossRef]
3. Anand, S.S.; Mathiyazaghan, R. Design and fabrication of voice controlled unmanned aerial vehicle. *IAES Int. J. Robot. Autom.* **2016**, *5*, 205–212.
4. Zheng, B.; Hu, J.; Zhang, G.; Wu, Y.; Deng, J. Analysis of noise reduction techniques in speech recognition. In Proceedings of the IEEE Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; pp. 928–933.
5. Ivanov, A.V.; Fazluktinov, P.S.; Kolesnev, V.A. Applying intelligent systems of speech recognition for optimizing the algorithm of noise reduction in audio records. *J. Phys. Conf. Ser.* **2020**, *1441*, 1–10. [CrossRef]
6. Tan, Z.H.; Varga, I. Network, distributed and embedded speech recognition: An overview. In *Automatic Speech Recognition on Mobile Devices and over Communication Networks*; Springer: London, UK, 2008; pp. 1–23.
7. Park, J.; Kim, J.; Oh, Y. Feature vector classification based speech emotion recognition for service robots. *IEEE Trans. Consum. Electron.* **2009**, *55*, 1590–1596. [CrossRef]
8. Lee, D.; Lim, M.; Park, H.; Kang, Y.; Park, J.; Jang, G.; Kim, J. Long short-term memory recurrent neural network-based acoustic model using connectionist temporal classification on a large-scale training corpus. *China Commun.* **2017**, *14*, 23–31. [CrossRef]
9. Wu, L.; Li, T.; Wang, L.; Yan, Y. Improving hybrid CTC/Attention architecture with time-restricted self-attention CTC for end-to-end speech recognition. *Appl. Sci.* **2019**, *9*, 4639. [CrossRef]
10. Ali, M.; Hameed, I.A.; Muslim, S.S.; Hassan, K.S.; Zafar, I.; Bin, A.S.; Shuja, J. Regularized urdu speech recognition with semi-supervised deep learning. *Appl. Sci.* **2019**, *9*, 1956.
11. Yang, X.; Tan, B.; Ding, J.; Zhang, J.; Gong, J. Comparative study on voice activity detection algorithm. In Proceedings of the IEEE International Conference on Electrical and Control Engineering, Wuhan, China, 25 June 2010; pp. 599–602.
12. Sun, Y.; Wang, R. Voice activity detection based on the improved dual-threshold method. In Proceedings of the IEEE International Conference on Intelligent Transportation in Big Data and Smart City (ICITBS), Halong Bay, Vietnam, 19–20 December 2015; pp. 996–999.
13. Pang, J. Spectrum energy based voice activity detection. In Proceedings of the IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 9–11 January 2017; pp. 1–5.
14. Dos SP Soares, A.; Parreira, W.D.; Souza, E.G.; de Almeida, S.J.; Diniz, C.M.; Nascimento, C.D.; Stigger, M.F. Energy-based voice activity detection algorithm using Gaussian and cauchy kernels. In Proceedings of the IEEE 9th Latin American Symposium on Circuits & Systems (LASCAS), Puerto Vallarta, Mexico, 25–28 February 2018; pp. 1–4.

15. Meier, S.; Kellermann, W. Artificial neural network-based feature combination for spatial voice activity detection. In Proceedings of the Interspeech, San Francisco, CA, USA, 8–12 September 2016; pp. 2987–2991.
16. Zazo Candil, R.; Sainath, T.N.; Simko, G.; Parada, C. Feature learning with raw-waveform CLDNNs for voice activity detection. In Proceedings of the the Interspeech, San Francisco, CA, USA, 8–12 September 2016; pp. 3668–3672.
17. Kim, J.; Hahn, M. Voice activity detection using an adaptive context attention model. *IEEE Signal Process. Lett.* **2018**, *25*, 1181–1185. [CrossRef]
18. Wang, Z.; Vincent, E.; Serizel, R.; Yan, Y. Rank-1 constrained multichannel Wiener filter for speech recognition in noisy environments. *Comput. Speech Lang.* **2018**, *49*, 37–51. [CrossRef]
19. Heymann, J.; Drude, L.; Haeb-Umbach, R. A generic neural acoustic beamforming architecture for robust multi-channel speech processing. *Comput. Speech Lang.* **2017**, *46*, 374–385. [CrossRef]
20. Wang, Z.Q.; Wang, D. All-neural multi-channel speech enhancement. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 3234–3238.
21. Xu, Y.; Du, J.; Dai, L.R.; Lee, C.H. A regression approach to speech enhancement based on deep neural networks. *IEEE Trans. Audio Speech Lang. Process.* **2015**, *23*, 7–19. [CrossRef]
22. Donahue, C.; Li, B.; Prabhavalkar, R. Exploring speech enhancement with generative adversarial networks for robust speech recognition. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5024–5028.
23. Bittu, K. Mean-median based noise estimation method using spectral subtraction for speech enhancement technique. *Ind. J. Sci. Tech.* **2016**, *9*. [CrossRef]
24. Martin, R. Spectral subtraction based on minimum statistics. In Proceedings of the IEEE European Signal Processing Conference, Edinburgh, UK, 13–16 September 1994; pp. 1182–1185.
25. Park, J.; Kim, J. Emotional information processing based on feature vector enhancement and selection for human—Computer interaction via speech. *Telecommun. Syst.* **2015**, *60*, 201–213. [CrossRef]
26. Ephraim, Y.; Malah, D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Trans. Signal Process.* **1984**, *32*, 1109–1121. [CrossRef]
27. Schwerin, B.; Pailwal, K. Using STFT real and imaginary parts of modulation signals for MMSE-based speech enhancement. *Speech Commun.* **2014**, *58*, 49–68. [CrossRef]
28. Malah, D.; Cox, R.; Accardi, A. Tracking speech-presence uncertainty to improve speech enhancement in non-stationary noise environments. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Phoenix, AZ, USA, 15–19 March 1999; pp. 201–204.
29. Kim, H.; Rose, R. Cepstrum-domain acoustic feature compensation based on decomposition of speech and noise for ASR in noisy environments. *IEEE Trans. Speech Audio Process.* **2003**, *11*, 435–446.
30. Michaely, A.H.; Zhang, X.; Simko, G.; Parada, C.; Aleksic, P. Keyword spotting for Google assistant using contextual speech recognition. In Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding, Okinawa, Japan, 16–20 December 2017; pp. 272–278.
31. Jeon, W.; Liu, L.; Mason, H. Voice trigger detection from LVCSR hypothesis lattices using bidirectional lattice recurrent neural networks. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 6356–6360.
32. Keshet, J.; Grangier, D.; Bengio, S. Discriminative keyword spotting. *Speech Commun.* **2009**, *51*, 317–329. [CrossRef]
33. Khalifa, S.; Hassan, M.; Seneviratne, A. Feasibility and accuracy of hotword detection using vibration energy harvester. In Proceedings of the IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Coimbra, Portugal, 21–24 June 2016; pp. 1–9.
34. Huang, Y.; Shabestary, T.Z.; Gruenstein, A.; Wan, L. Multi-microphone adaptive noise cancellation for robust hotword detection. In Proceedings of the Interspeech, Graz, Austria, 15–19 September 2019.
35. Ge, F.; Yan, Y. Deep neural network based wake-up-word speech recognition with two-stage detection. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 2761–2765.
36. Varga, A.; Steeneken, H.J. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech Commun.* **1993**, *12*, 247–251. [CrossRef]

37. Hirsch, H.G.; Pearce, D. The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy conditions. In Proceedings of the International Conference on Spoken Language Processing, Beijing, China, 16–20 October 2000; pp. 29–32.
38. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.