ICA 2019 AACHEN

PROCEEDINGS of the
23rd International Congress on Acoustics

9 to 13 September 2019 in Aachen, Germany

Intelligent background sound event detection and classification based on WOLA spectral analysis in hearing devices

Feifan LAI[(1)], Kaibao NIE[(2)]

[(1)]University of Washington, Bothell, WA, USA, feifanlai1993@gmail.com

[(2)]University of Washington, Bothell, WA, USA, niek@uw.edu

ABSTRACT

The purpose of this paper is to build a system model which can automatically separate background sound from noisy speech signal, and then classify the background into predefined event categories. This paper proposed to use weighted overlap-add algorithm (WOLA) for feature extraction and neural network (NN) for sound event detection, with recordings of 11 realistic background noise scenes (cafe, station, hallway …), mixed with human speech at SNR of 5 dB. In our approach, an energy waveform trough detection algorithm is used to retrieve important background sound information. Then the WOLA algorithm is used to extract spectral features by transforming each fraction of time domain signal into frequency domain data represented in 22 channels. Moreover, a feed-forward neural network with one hidden layer is used to effectively recognize each event's diverse spectral feature pattern, and then produces classification decisions in the form of confidence values. The overall detection accuracy has achieved 96%, while the event 'hallway' has the lowest detection rate at 85%. Moreover, this detection algorithm could improve noise reduction in hearing devices by applying distinct compensation gains, which will attenuate the noise dominated frequency bands for each particular predefined event.

Keywords: WOLA, hearing devices, sound event detection

1. INTRODUCTION

Hearing aid and cochlear implant are two main types of hearing devices that can help people with hearing loss regain partial hearing ability. People with a hearing aid or cochlear implant generally still have difficulty in understanding speech in background noise. Smart sound processing in detecting and classifying background events can potentially improve their speech understanding in adverse listening environments.

Efficient ways of separating background noise from complex sounds is a crucial component for smart sound processing. Speech pause detection is one useful method for noise background separation. The pause detection algorithm tracks the power envelops of the speech + noise signals and marks the minima points as 'pauses' [15]. In this study, a simplified 'trough detection' algorithm is developed on the basis of the speech pause detection algorithm.

The weighted-overlap add (WOLA) analysis is an efficient and robust tool for processing audio signals and it is widely available on hearing device DSP (digital signal processing) platforms. WOLA is an analysis and synthesis technique which breaks down a piece of continuous audio signal into separated small segments for feature extraction process. Brennan et al. [12] have built an adaptive filtering system using WOLA in an ultra-low-power DSP. Moreover, Sheikhzadeh et al. [10] proposed a delayless method for adaptive filtering through subband adaptive filters (SAF) systems based on WOLA, taking advantage of WOLA's block processing feature which doesn't involve long FFT or IFFT operations. In another study by Vicen-Bueno et al. [11], the WOLA filterbank was used in a modified LMS-based (Least Minimum Square) feedback-reduction subsystems in digital hearing aids.

Spectral subtraction techniques involve the use of FFT (Fast Fourier Transform) or filter banks. It estimates the power spectrum of noise-free signal by subtracting the overall signal power spectrum with the noise-only power spectrum. Mwema and Mwangi [2] proposed an algorithm based on the

spectral subtraction method to eliminate the Gaussian white noise. In this paper, the spectral subtraction technique is used and the event feature gain pattern is applied to attenuate the noise frequency channels.

Neural network applications in DSP has been gaining attention in recent years, due to the increased processing power in low-cost/low-power DSPs. The neural network running efficiency is exploited by Behan et al. [3]. They have performed ways to accelerate Integer Neural Networks (INN) on low cost DSPs by using DSP instructions. In another study by Namjin et al. [9], a DSP-Based Hierarchical Neural Network is used for classification of 11 modulation signals by analyzing on 31 statistical signal features. With similar idea, in this study, neural network is used to classify 11 sound events based on 22-channel spectrum features.

The proposed intelligent background sound event detection system aims to classify a given sound signal into one of the preset scenes. The structure of this paper is as follows. The trough detection algorithm used for separating background noise from speech audio signals is presented in Section 2. The feature extraction procedure, including the use of WOLA for sound analysis and synthesis, as well as the smoothing of estimated noise templates, are described in Section 3. In Section 4, the application of neural networks in sound event classification and its detailed structure design are discussed, as well as the evaluation of the detection accuracy derived from the testing dataset. The application of noise reduction gain patterns and results are discussed in Section 5. And lastly, the conclusions and future work are discussed in Section 6.

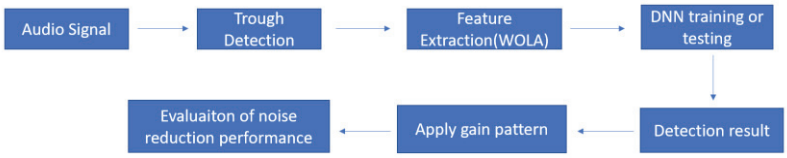


Figure 1. Framework of the main signal processing procedures.

2. TROUGH DETECTION ALGORITHM

In this study, sound event refers to a set of environmental noises in 11 different scenes. The background noise audio files are downloaded from the Creative Commons website. They include 'cafe', 'cafeteria', 'hallway', 'kitchen', 'living', 'meeting', 'office', 'resto', 'square', 'station', and 'washing'. Each event contains 16 files, and each file lasts 5 minutes long. The ideal type of background noise signals for detection should be stationary with a constant volume. Events of ‘cafe’, ‘cafeteria’, ‘living’, ‘meeting’, and ‘resto’ are presumed to be relatively more difficult for detection, since they contain varying sound elements. In this study, the first 4 minutes of each file is used for training, while the last 1 minute of each file is reserved for detection testing.

Events	Number of files	Length of each file	Total length of audio	Major sound elements
café	16	5 min	80 min	chatting, tableware, footsteps
cafeteria	16	5 min	80 min	chatting, tableware, footsteps
hallway	16	5 min	80 min	footsteps, door
kitchen	16	5 min	80 min	water streaming, dishes
living	16	5 min	80 min	TV music, newspaper
meeting	16	5 min	80 min	consersation, desk
office	16	5 min	80 min	desk, typing
resto	16	5 min	80 min	chatting, tableware, footsteps
square	16	5 min	80 min	crowded people, chatting, birds
station	16	5 min	80 min	vehicles, white noise
washing	16	5 min	80 min	water streaming

Table 1: Sound file list.

The speech file contains a piece of human speech consisting of 3 sentences and lasting about 6 seconds. The audio signals used in this study are mixed audios generated by adding event noise signals to human speech signals with SNR at +5 dB and a sampling rate of 24 kHz. To generate a mixed audio at desired SNRs, the formula below is used:

$$S_{mixed} = S_{speech} + S_{noise} \times \frac{RMS_{speech}}{RMS_{noise}} \times 10^{\frac{-SNR}{20}} \quad (1)$$

Background extraction is the crucial part and also the first stage for this system. The objective of the proposed ‘trough detection’ method is to temporally locate the background noise data from the mixed signal. To achieve that, the mixed audio signal is first converted to a sound energy waveform,

by applying a high pass filter (preemphasis) and half-wave rectifier to the audio signal, and transforming every 256 original audio data samples (equivalent to roughly 21.5 ms of signal) to one RMS (Root Mean Square) sample.

Then the 'trough detection' algorithm could be applied to energy waveforms with the following procedure steps:

1. From the starting point, follow the curve of energy waveform.
2. When an energy trough is found, mark it as 'energy trough'.
3. Force the trough tracking curve to increase linearly at a preset time constant starting from this 'energy trough'.
4. Stop the increase where it meets the audio energy waveform, then follow the curve of energy waveform again.
5. Again, when an energy trough is found, mark it as 'energy trough'.
6. The same process repeats itself by looping step 3 to step 5, until the end of energy waveform is met.

The gradually ascending line represents the RC (Resistor Capacitor) response time. In a Resistor Capacitor circuit, the RC response time is the time it takes to charge the capacitor from 0 to 63% of its max value. And the RC response time determines the time constant (tau-τ), defined by resistance R multiplies capacitance C. A similar idea is used in this trough detection algorithm. The slope angle of the line is determined by the tau value. The time constant was set based on the speech syllabic rate to force the trough tracking curve staying on the noise floor.

A reliable trough is a 'global' trough which is considered a low energy spot through the entire timeline of energy waveform. The use of RC response line prevents extracting an unreliable 'local' trough. 'Local' troughs are low energy spots in small local segments on energy waveform. They generally have lower energy than adjacent troughs but still have higher energy than 'global' troughs. In Figure 2, a 'local' trough is marked by the red cross. A higher energy trough indicates that it probably contains speech data samples. So 'local' troughs are not wanted, as they don't represent noise-only samples.

The last step is to extract these energy troughs. Energy troughs contain background noise-only data. Because human speech usually has periodic higher energy, while the background has relatively constant and lower energy.

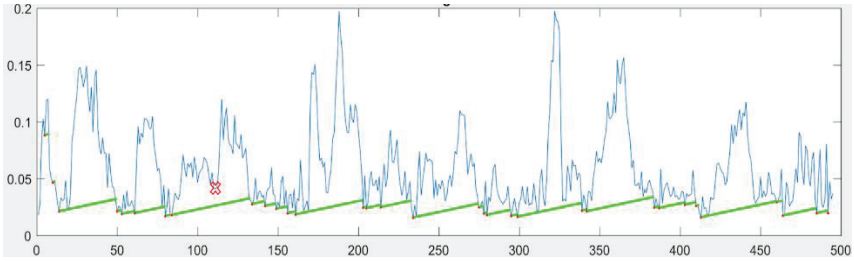


Figure 2. Demonstration of the application of the trough detection algorithm. The energy waveform is derived from a mixed audio signal of event 'station' at SNR = +5 dB. The x axis is the number of energy waveform samples. The green line is the trace of RC response line. The red dots are troughs detected, and they represent the background noise-only samples to be extracted.

3. FEATURE EXTRACTION

The weighted overlap-add algorithm is widely useful in analyzing audio signal. WOLA analyses the time domain signal block by block, making it an ideal method for real time processing devices. Schuster and Ansorge [8] have experimented on WOLA's ability to filter out background noise in a noise canceling system and achieved de-noised speech.

The WOLA filterbank is a low-delay, highly efficient implementation of over-sampled generalized DFT (Discrete Fourier Transform) system [13, 14]. For WOLA analysis, the first step is to shift R input samples at a time to the buffer with frame length L [6]. Note that when odd stacking is used, the input signal must be flipped in sign every N samples [7]. And then, this buffer will be processed by an analysis window with also length L. Lastly, this L-sample frame will be folded into N-sample blocks for Fast Fourier Transform (FFT) process to produce spectrum output in complex numbers. Its mathematical framework is defined as [5]:

$$X_k(sR) = \sum_{m=-\infty}^{\infty} h(sR-m)x(m)W_M^{-mk} \quad (2)$$

where x(m) is a time domain signal sampled every R samples in time m, $W_M = e^{\frac{j2\pi}{M}}$, M is the number of frequency samples, k is the discrete frequency index, $h(m)$ is the analysis window, and s is the time index of the short-time transform at the decimated sampling rate (decimated by the integer factor R).

WOLA synthesis is like the reverse of WOLA analysis. It applies inverse Fourier transformation to segments of short-time Fourier spectrum and sum the results with overlap to obtain audio signal in time domain. Its main characteristic is to transform frequency spectrum data back to a finite length of time domain signal. Its mathematical framework is defined as [5]:

$$\hat{X}(n) = \sum_{s=-\infty}^{\infty} f(n-sR')\frac{1}{M}\sum_{k=0}^{M-1}\widehat{X_k}(sR')W_M^{nk} \quad (3)$$

where $\hat{X}_k(sR')$ is a discrete short-time spectrum, output $\hat{x}(n)$ is considered as sampled every R' samples in time n, and $W_M = e^{\frac{j2\pi}{M}}$, M is the number of frequency samples, k is the discrete frequency index, $f(n)$ is the synthesis window. and s is the time index of the short-time transform at the decimated sampling rate (decimated by the integer factor R').

The operation of WOLA filterbank is determined by a number of parameters: Analysis window size (La), Synthesis window size (Ls), Input block step size (R), and FFT size (N). In our experiments, La is set to 512, Ls is 256, N is 256, R is 64, and odd stacking is used to produce reasonable spectral and temporal resolutions.

Next, each extracted energy trough is processed through WOLA analysis and returns 128 complex numbers. All complex numbers are transformed to real numbers by magnitude calculation to represent the value of 128 bins in frequency domain. After 128 bin features are created, they are converted into 22-channel feature curves by bin merging. Because in cochlear implants, only up to 22 channels of stimulating signals are generated to mimic normal hearing. Since the speech signal generally occupies the low-mid frequency, it is important to reserve a higher resolution of data in 0.1 kHz ~ 3 kHz frequency range.

128 bin number	1~5	6~7	8~9	10~11	12~13	14~15	16~17	18~19	20~21	22~25	26~29
22 bin number	1	2	3	4	5	6	7	8	9	10	11
128 bin number	30~33	34~37	38~43	44~49	50~57	58~65	66~75	76~85	86~97	98~111	112~128
22 bin number	12	13	14	15	16	17	18	19	20	21	22

Table 2: WOLA bins that are combined into 22 channels.

Each event has its own signature feature curve shape. For better observation, the plots of four events' averaged feature curve out of roughly 1500 feature curves are shown below. Those plots give valuable information about how this intelligent detection system 'sees' these events. The distinguishable patterns of each event create the opportunity for machine recognition.

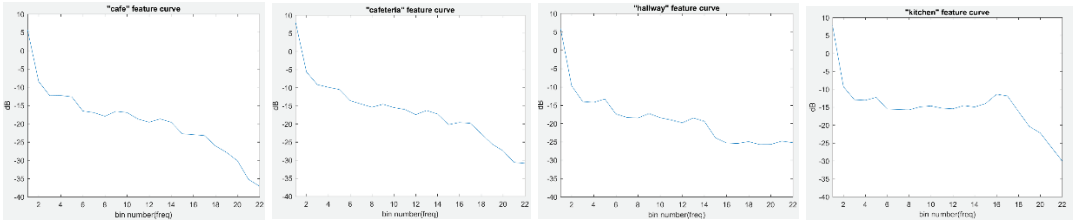


Figure 3. A direct comparison of 4 events' averaged feature curves, including events 'cafe', 'cafeteria', 'hallway', 'kitchen'. The horizontal axis stands for frequency channels, and the vertical axis is magnitude in dB.

4. NEURAL NETWORK DETECTION

Neural network's automatic structure learns the highly nonlinear mapping between the input and the output directly from data, therefore reducing the need to find human-crafted intermediate representations [12]. A Feed-forward neural network with 1 hidden layer is used in this study. Also, the training function of Levenberg-Marquardt is used. In our experiment, the neural network accepts 22 input values from 22 channels, and outputs 11 values representing the classification confidence for the 11 pre-defined events.

Figure 4. The neural network 1-layer structure.

Each neuron in hidden layer and output layer forms a web structure, as described in this formula:

$$o = f(i * w + b) \tag{4}$$

where 'i' denotes the input to the neuron, 'o' denotes the output of the neuron, 'w' is weight, 'b' is bias, and 'f' is the transfer function.

First of all, a neuron takes an input either from feature curves or from the output of previous layer. The input is multiplied by a weight value and then added with a bias value. After that, a non-linear Log-Sigmoid transfer function is applied to form the neuron output.

The training set is a matrix with a size of 22 rows and around 300,000 columns, which represents the 22 bins values of around 300,000 feature curves extracted from all the test audio files. Another matrix of targeted result is also stored, which has 11 rows and same number of columns. This target matrix tells the neural network which feature curve belongs to which event. During the training procedure, the assigned values for weights and bias would be adjusted to approach producing output results similar to target matrix.

After neural network is built, a 5 seconds time frame is sufficient to give a detection classification result. The output from the neural network will give 11 values, representing the confidence level (likelihood) of the 11 events. And the event with the largest confident value will be selected as the detected event.

Detection Averaged Confidence Matrix for all events

Detection Results	café	cafeteria	hallway	kitchen	living	meeting	office	resto	square	station	washing
café	0.9237	0.0003	0.0004	0.0027	0.0009	0	0.0056	0.0031	0.0447	0.0414	0.0206
cafeteria	0.0113	0.6966	0.0019	0.0364	0.0195	0.004	0.0499	0.1187	0.3614	0.0084	0.0762
hallway	0.0035	0.0017	0.6988	0.008	0.0442	0.3241	0.008	0.0047	0.0069	0.0034	0.0035
kitchen	0.0013	0.0588	0.0032	0.9886	0.0047	0.0035	0.0068	0.0368	0.0107	0.0048	0.0071
living	0.0004	0.001	0.1268	0.0079	0.9942	0.109	0.0036	0.0024	0.0009	0.0005	0.001
meeting	0.0002	0.002	0.1772	0.0033	0.0691	0.5676	0.0206	0.0021	0.0111	0.002	0.0041
office	0.0012	0.078	0.0021	0.006	0.0166	0	0.9354	0.0018	0.0718	0.0035	0.1074
resto	0.0083	0.1623	0.0026	0.0103	0.0154	0.0076	0.0035	0.8457	0.0454	0.004	0.0131
square	0.0293	0.0181	0.0031	0.0064	0.0031	0.0043	0.0279	0.0127	0.3692	0.1252	0.2265
station	0.0052	0.0156	0.001	0.0228	0.0015	0.0061	0.0036	0.0172	0.1307	0.7993	0.0177
washing	0.0439	0.0015	0.0003	0.0044	0.0033	0.0011	0.0013	0.0124	0.056	0.0184	0.762

Table 3: Confidence Matrix shows the overall averaged confidence values for all 11 events. Each column is one event under detection, and the rows are the confidence level of 11 events.

Table 3 lists out the averaged confidence values for all 11 events. The confidence value is the output from neural network ranging from 0 to 1(0% to 100%). After each detection, one confidence level will be generated for each event, representing the likelihood to be recognized as that event. It gives information about the similarity of two events' feature curves from the detection system's perspective. For example, it could be found that 'meeting' is recognized as similar to 'hallway' by observing that 'meeting' has a 0.1772 confidence value to 'hallway', at the same time, 'hallway' has a 0.3241 confidence value to 'meeting'.

Detection Confusion Matrix for all events

Detection Results	café	cafeteria	hallway	kitchen	living	meeting	office	resto	square	station	washing
café	98%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
cafeteria	0%	96%	0%	1%	0%	0%	2%	4%	3%	0%	0%
hallway	0%	0%	85%	0%	0%	2%	0%	0%	0%	0%	0%
kitchen	0%	0%	0%	99%	0%	0%	0%	0%	0%	0%	0%
living	0%	0%	6%	0%	99%	0%	0%	0%	0%	0%	0%
meeting	0%	0%	9%	0%	1%	98%	0%	0%	0%	0%	0%
office	0%	0%	0%	0%	0%	0%	97%	0%	0%	0%	1%
resto	0%	4%	0%	0%	0%	0%	0%	96%	0%	0%	0%
square	0%	0%	0%	0%	0%	0%	1%	0%	96%	1%	4%
station	0%	0%	0%	0%	0%	0%	0%	0%	1%	99%	0%
washing	2%	0%	0%	0%	0%	0%	0%	0%	0%	0%	95%

Table 4: Confusion Matrix shows the overall detection accuracy for all 11 events. Each column is one event under detection, and the rows are the recognition rates in percentage for 11 events.

Table 4 lists out the overall detection accuracy for all 11 events. It provides a general view of the recognition correctness rate for each event. It also indicates which event is misclassified if error occurs.

Table 3 and Table 4 are related but not the same. The averaged confidence levels could not be directly translated into detection accuracy. In other words, if one event has a very low averaged confidence value of itself, it couldn't imply its detection accuracy is low. For example, event 'square' has a 0.3692 averaged confidence value to be detected correctly, and a 0.3614 averaged confidence value to be incorrectly classified as 'cafeteria'. By reading the confidence values, it looks like event 'square' will often be recognized as 'cafeteria', as they have similar averaged confidence values. However, the confusion matrix shows that 'square' has a high 96% accuracy, and only a 3% chance to be wrongly detected as 'cafeteria'. The reason is that although the two events have similarly low confidence levels, the confidence values of 'cafeteria' is still smaller than that of 'square' in most of the time. The two main reasons for misclassification are: similar feature curves between two events; variance of the feature curves of one event.

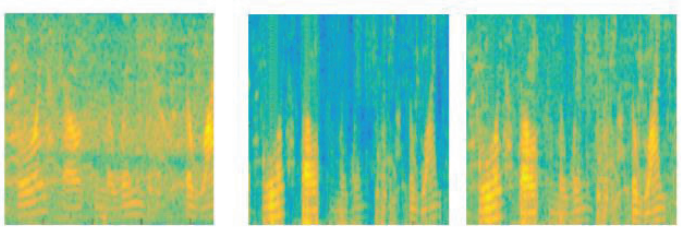
5. NOISE REDUCTION WITH GAIN PATTERN

Noise reduction is one important real field application to take advantage of the automatic event detection results. The de-noise gain patterns for each event could be derived from the events' averaged feature curve patterns. A meaningful event gain pattern with the attenuation range of 0 dB to -30 dB can be derived from flipping-over the event averaged feature curve, and then apply normalization and pre-emphasis. A de-noised gain pattern could be easily applied to the frequency domain audio data after WOLA analysis but before WOLA synthesis. This de-noising gain pattern attenuates the noise dominating frequency channels.

The noise reduction performance is evaluated preliminarily by listening to the audios before/after de-noising, and visually comparing the spectrogram of input/output audio signals. The result shows that after applying the gain pattern, the output audio sounds noticeably less noisy, with the speech more popped out. But the event gain pattern won't react to sudden or periodic changes in background noises. It is most effective for elimination of stationary noise with constant volume level and similar characteristics.

Another method to generate de-noise gain pattern is using the instant trough background noise data [4]. In this approach, the last four troughs would be extracted and fed into a FIFO (First In First Out) to generate an averaged instant feature curve, which would be applied to same processes of flip-over, normalization, and pre-emphasize to derive the instant gain pattern for the current time moment. Since this instant gain pattern is updated every time when a new trough is detected, it is adaptive to the dynamically changing background noise. Now even the abrupt noise could be deeply suppressed. However, the instant gain doesn't update itself if no new trough is detected. Then the instant gain pattern has too little attenuation on the noise or mistakenly attenuates human speech. Through preliminary testing, it is found that the output audio's speech volume goes up and down from time to time, caused by the abrupt changes of the instant gain pattern. Also, the speech doesn't sound smooth, as the first and last syllables of a word are usually cut off.

To overcome the limitations of both the event gain pattern and instant gain pattern, a combination of them with an adjustable weighting ratio is tested in the experiment. The test results have proven that the combined gain pattern generates output audio with less noise over the course and smoother speech transitions between syllables. By observation of its spectrogram, it's shown that output audio has improved noise reduction ability (larger blue area) compared to Figure 5. (a), and meanwhile it has smoother speech transitions (less gaps in low frequency yellow area) compared to Figure 5. (b). Therefore, the combined gain pattern can potentially improve de-noising results.



(a) (b) (c)

Figure 5. These three images are zoom-in comparison between three types of gain patterns applied. Image (a) is the spectrogram result of using the event gain pattern. Image (b) is the spectrogram result of using the instant gain pattern. Image (c) is the spectrogram result of using the combined gain pattern at ratio of 1:1. Image (a) preserves and enhances all the details in human speech. But it has horizontal spectral gaps, since its gain pattern remains the same over time. And it's less effective in reducing noise which doesn't match the event gain pattern. Image (b) could greatly reduce noise. It applies adaptive instant gain pattern from the background noise in real time. But it creates temporal (vertical) speech gaps, which causes the speech stuttering. Image (c) combines the detection results of the two gain patterns. It is more efficient in noise reduction, while retaining the smoothness of human speech.

6. CONCLUSIONS AND DISCUSSION

In this study, a novel approach for sound event detection in realistic environments using WOLA and a feedforward neural network was proposed. The trough detection algorithm has been proven to be able to accurately extract background sections. WOLA is capable of transforming those background time domain sections into frequency spectrum (feature) curves for gain application. Then these feature curves are used as training data for neural network. After the neural network is built, mixed audio signals can be fed into neural network for event classification. This model could also contribute to background noise reduction by attenuating the targeted background noise frequency regions. A gain pattern could be derived by flipping the event feature curve. This gain pattern is later applied back to the mixed audio signal in complex domain for noise reduction purpose. WOLA synthesis is used to transform the frequency domain data back to continuous time domain audio. Preliminary evaluations were carried out for a noise reduction efficiency test, and the noise suppression performance for pre-defined events is improved.

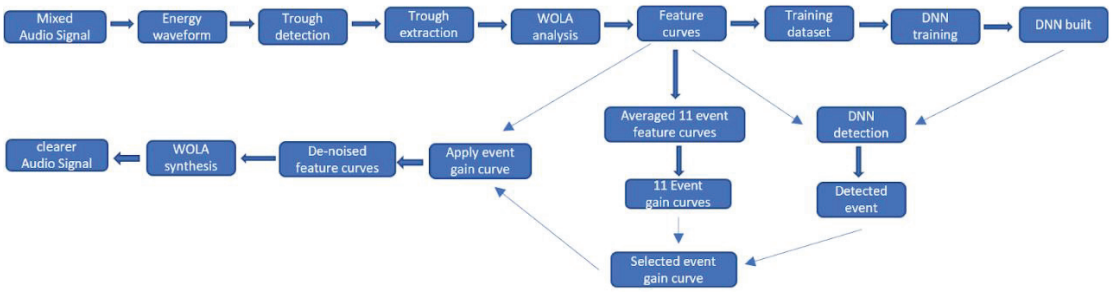


Figure 6. block diagram of the intelligent event detection and noise reduction systems of this paper.

For future work, a larger training dataset with more event signals could be used for training to further understand the full capability of this system. Future work could be made to test this system on a real hearing device. Furthermore, this system could be applied to not only noise detection, but also smart sound recognition, such as keyword spotting.

REFERENCES

1. Strom KE. The HR 2006 Dispenser Survey. Hear Rev. 2006;13:16-39.
2. W. N. Mwema and E. Mwangi, "A spectral subtraction method for noise reduction in speech signals," Proceedings of IEEE. AFRICON '96, Stellenbosch, South Africa, 1996, pp. 382-385 vol.1. doi: 10.1109/AFRCON.1996.563142.
3. Behan, Thomas & Liao, Zaiyi & Zhao, Lian & Yang, Chunting. (2008). 1 Accelerating Integer Neural Networks On Low Cost DSPs. International Journal of Intelligent Systems - IJIS.

4. K. Nie *et al*., "A WOLA-Based Real-Time Noise Reduction Algorithm to Improve Speech Perception with Cochlear Implants," *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Honolulu, HI, USA, 2018, pp. 951-954. doi: 10.23919/APSIPA.2018.8659579.
5. R. Crochiere, "A weighted overlap-add method of short-time Fourier analysis/Synthesis," in IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 28, no. 1, pp. 99-102, February 1980. doi: 10.1109/TASSP.1980.1163353.
6. Ahadi, S.M., Sheykhzadeh, H., Brennan, R.L., & Freeman, G.H. (2004). A Weighted Overlap Add-based Front-end for Speech Recognition.
7. R. L. Brennan, R. Abutalebi and H. Sheikhzadeh, "Adaptive filtering using a highly oversampled weighted overlap-add filterbank in an ultra low-power system," Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002., Pacific Grove, CA, USA, 2002, pp. 806-810 vol.1. doi: 10.1109/ACSSC.2002.1197290.
8. G. Schuster and R. Ansorge, "WOLA noise cancelling performance," 2008 16th European Signal Processing Conference, Lausanne, 2008, pp. 1-5.
9. F. Frey, R. Elschner, C. Kottke, C. Schubert and J. K. Fischer, "Efficient real-time implementation of a channelizer filter with a weighted overlap-add approach," 2014 The European Conference on Optical Communication (ECOC), Cannes, 2014, pp. 1-3. doi: 10.1109/ECOC.2014.6964101.
10. H. Sheikhzadeh, R. L. Brennan, Z. Khan and K. R. L. Whyte, "Low-resource delayless subband adaptive filter using weighted overlap-add," 2005 13th European Signal Processing Conference, Antalya, 2005, pp. 1-4.
11. R. Vicen-Bueno, A. Martinez-Leira, R. Gil-Pita and M. Rosa-Zurera, "Modified LMS-Based Feedback-Reduction Subsystems in Digital Hearing Aids Based on WOLA Filter Bank," in IEEE Transactions on Instrumentation and Measurement, vol. 58, no. 9, pp. 3177-3190, Sept. 2009. doi: 10.1109/TIM.2009.2017150.
12. Y. Bengio, “Learning deep architectures for AI,” Foundations and trends in Machine Learning, vol. 2, no. 1, pp. 1–127, 2009.
13. R.E. Crochiere and L. R. Rabiner, Multirate digital signal processing, Prentice-Hall, 1983.
14. R. Brennan and T. Schneider, “A flexible filterbank structure for extensive signal manipulations in digital hearing aids”, Proc. IEEE Int. Symp. Circuits and Systems, pp.569–572, 1998.
15. M. Marzinzik and B. Kollmeier, "Speech pause detection for noise spectrum estimation by tracking power envelope dynamics," in IEEE Transactions on Speech and Audio Processing, vol. 10, no. 2, pp. 109-118, Feb. 2002. doi: 10.1109/89.985548