ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika

ISSN(p): 2338-8323 | ISSN(e): 2459-9638 | Vol. 9 | No. 4 | Halaman 856 - 866
DOI : http://dx.doi.org/10.26760/elkomika.v9i4.856 Oktober 2021

HiVAD : A Voice Activity Detection Application Based on Deep Learning

MUHAMMAD HILMI FARIDH, ULIL SURTIA ZULPRATITA

Universitas Widyatama Bandung, Indonesia
Email: hilmi.faridh@widyatama.ac.id

Received 17 Mei 2021 **|** *Revised* 2 Juni 2021 **|** *Accepted* 15 Juni 2021

ABSTRAK

Dalam tulisan ini, deteksi aktivitas suara disajikan pada smartphone secara real-time dengan jaringan saraf konvolusional. Pengurangan waktu komputasi adalah masalah dari studi sebelumnya. Meskipun telah menggunakan pendekatan machine learning, masih banyak kekurangan dari penelitian sebelumnya. Citra sinyal suara dihasikan oleh spektrogram energi log-mel, kemudian citra sinyal suara diinputkan ke dalam deep learning CNN untuk mengklasifikasi suara manusia dan derau. HiVAD mengungguli persentase metode VAD lainnya yaitu G729B, Sohn, dan RF dari hasil tes yang ditunjukkan dengan akurasi rata-rata SHR sebesar 15,89%, 28,98%, 42,13% pada tingkat 0dB, 8,67%, 16,29%, 17,63% pada tingkat 5 dB, dan 1,35%, 7,72%, 5,14% pada tingkat 10 dB. Selain itu, mekanisme Multi-threading memungkinkan komputasi yang efisien untuk waktu secara real-time. Penelitian ini menunjukkan bahwa arsitektur CNN pada HiVAD secara signifikan meningkatkan akurasi deteksi aktivitas suara.

***Kata kunci**: aplikasi VAD, voice detection, deep learning, CNN*

ABSTRACT

In this paper, the detection of sound activity is presented on smartphones in real-time with convolutional neural networks. Reduced computing time is a problem from previous studies. Despite the use of machine learning approaches, there are still many shortcomings from previous research. A log-mel energy spectrogram narrates the sound signal image. Then the sound signal image is inputted into CNN's deep learning to classify the human voice and noise. HiVAD outperformed the percentage of other VAD methods, namely G729B, Sohn, and RF from the test results shown with an average SHR accuracy of 15.89%, 28.98%, 42.13% at 0dB, 8.67%, 16.29% ,17.63% at 5 dB, and 1.35%, 7.72%, 5.14% at 10 dB. In addition, the Multi-threading mechanism enables efficient computing for real-time. This study shows that CNN's architecture on HiVAD significantly improves the accuracy of sound activity detection.

***Keywords**: VAD App, voice detection, deep learning, CNN*

1. INTRODUCTION

Voice Activity Detection (VAD) is a speech processing method in which the present or not human voice is found. VAD characteristics are based on sound and noise. In the process of voice communication, some users will inevitably encounter various interferences that will interfere with the received sound signal, the receiver, in the end, is not the original sound signal, but the sound signal mixes with the noise **(Yang, et al, 2010)**.

The deep learning approach already features the detection of sound activities that can try more efficiently. However, that approach has a very long computational time and creates obstacles in its utilization in real-time sound processing. A real-time system is a program whose operation precision depends on the results of computing logic and time **(Chandra, 2018)**.

Hypothetically, the eyes are much smarter and faster compared to the ears; If the frequency of sound can be converted into images in a certain way and sent to the eyes to be distinguished, we can understand the greater frequency range **(Rishi, 2019)**. CNN is a category of code that is widely used in image data. CNN can identify and categorize features in pictures. The findings of the spectrogram log-mel representation will be built into CNN's architecture to minimize computation time and push the resulting compute higher **(Krizhevsky, et al, 2017)**. The log-mel spectrogram designed into CNN's architecture will isolate noise signals from pure human sound signals **(Sehgal & Kehtarnavaz, 2018)**.

VAD has been developed previously using machine learning techniques. In **(Dong, et al, 2002)** use the G729.B feature with a Support Vector Machine (SVM). In **(Ramírez, et al, 2006)** use Signal-To-Noise Ratio (SNR) with SVM classification. **(Jo, et al, 2009)** use the probability ratio of statistical models together with SVM classification. **(Saki & Kehtarnavaz, 2016)** VAD was created with subband technology along with Random Forest (RF) classification. The use of deep learning models in VAD has already been reported in research journals. **(Zhang & Wu, 2013)**, for example, utilizing a set pitch is one of the functions, DFT, Cepstral Mel-Frequency Coefficient (MFCC), Linear Predictive Code (LPC), Relative-Spectral Perceptual Linear Prediction Analysis (RASTA-PLP), and Amplitude Modulation Spectrogram (AMS) with the aid of a deep learning model of beliefs. **(Thad & Keir, 2013)** use the 13-dimensional Perceptual Linear Prediction (PLP) feature along with Recurrent Neural Network (RNN). In **(Thomas, et al, 2014)**, A Convolutional Neural Network (CNN) was used in conjunction with a log-mel spectrogram with regression and its activation coefficient. Until sound movement detection, **(Obuchi, 2016)** used an Augmented Statistic Noise Suppression (ASNS) to increase VAD accuracy. In this VAD, energy log Mel filterbank vector features are integrated into the classification using decision trees, SVMs, and CNN's. Much research on VAD, but some things do not pay much attention, like real-time computing time. In this study, the authors proposed a VAD algorithm applied to a smartphone application to detect voice activity using Convolution Neural Network's deep learning approach.

2. METHODS

HiVAD is a deep learning-based voice activity technology application developed to separate noise and pure human sound. HiVAD is implemented using several components so that an application can be run in real-time.

2.1 Energy Log-Mel Spectogram

Log-Mel Spectogram is an audio representation as an image that represents the temporary strength of an acoustic indication in the mel frequency gage. A log-mel spectrogram is formed

by a Frequency-To-Mel Spectral Coefficient (MFSC) over a while. Similar to MFCC but DCT calculation taken from MFCC **(Sehgal & Kehtarnavaz, 2018)**.

The sound is split into short frames with 20-40 ms to measure the MFSC, and This is done considering the quasi-stationary sound signal. Tests conducted for a sufficient period will show the characteristics of stationary sound signals. But when done over a more extended period, the features of the sound signal will continue to change according to the spoken word **(Sehgal & Kehtarnavaz, 2018)**.

A triangular overlapping filter bank N is required to calculate the MFSC because to limit the spectrogram. The higher and lower frequencies must be determined to obtain the N+2 frequency distance among the lesser and upper frequencies. Mel filter bank distance is shown in Equation (1) and (2).

$$\hat{f}(n) = \frac{(K+1)*B^{-1}(\hat{m}(n))}{f_s}, \quad n = 0 \ldots N+1 \tag{1}$$

$$H_n(k) = \begin{cases} 0, & k < \hat{f}(n-1) \\ \dfrac{k - \hat{f}(n-1)}{\hat{f}(n) - \hat{f}(n-1)} & \hat{f}(n-1) < k \leq \hat{f}(n) \\ \dfrac{\hat{f}(n+1) - k}{\hat{f}(n+1) - \hat{f}(n)} & \hat{f}(n) < k \leq \hat{f}(n+1) \\ 0, & k > \hat{f}(n+1) \end{cases}$$

$$k = 1 \ldots K/2$$

$$n = 1 \ldots N \tag{2}$$

In the mel field, F is a series with N+2 frequencies bin functions from the same filter, and H represents the node n filter amplitude at the bin k frequency. The filter bank is then multiplied by the FFT's estimated power set. As shown in Equation (3), the MFSC is determined by adding the values of each filter and taking the logs of each sum.

$$MFSC(n) = \log\left(\sum_{k=0}^{K} H_n(k) * |F(k)|^2\right) \tag{3}$$

They were added to obtain an image of N x B, where B indicates the percentage of frames considered in the continuum after discovering the MFSC N coefficient. It then generates a graph known as the log-mel energy curve. After that, CNN receives the data. Figure 1 depicts the procedures for obtaining log-mel spectrum capacity.

Using log-mel energy continuum representations as inputs to CNN, parts or sections of a fragmented sound pulse of individual words may have been distinguished with parts or segments without speech content or from sheer noise. The red/yellow portion of the log-mel energy continuum shows the presence of voice, while the remainder of the picture appears green/blue as background noise. Figure 2 llustrates this.

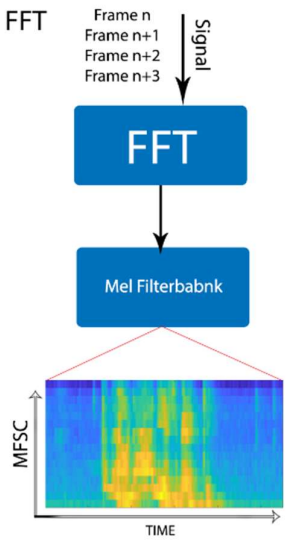

Figure 1. In The Diagram of Spectrum Image Formation

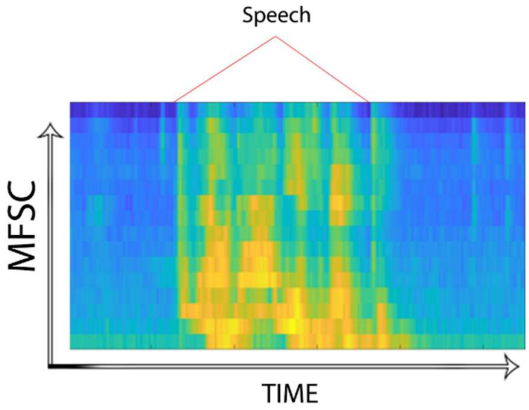

Figure 2. An Image Of A Log-Mel Energy Spectrogram

2.2 Classification

The Convolutional Neural Network (CNN) is used for classification and decision-making. CNN is still being used to deploy VAD **(Thomas, et al, 2014)**. This Neural Network handles images as inputs with hidden layers, CNN running convolution and convergence operations, and linked layers.

Via a kernel that can be trained with non-linear activation, the convolution layer can derive local knowledge from input/matrix images. The input space is repeated with this kernel. Feature maps may be max-pooled or convolution with longer steps to minimize their resolution before being placed into the next convolution sheet. The layer is used to merge the data from the final state of the convolution layer and thereby characterize that comprehensive data to use a final non-linear output until it is fully connected. In this case, the activation function is a feature map representing the probability of two classifications: pure sound or loud sounds only and voice in sound **(Sehgal & Kehtarnavaz, 2018)**.

Figure 3 depicts the behavior of Convolution Neural Networks. The N x B log-mel energy curve is described here, with N denoting the coefficient value and B denoting the number of pixels

used as spectrum inputs. In some instances, when it comes to gathering temporal data, the value of B is considered to be higher than the value of N. However, in Increasing computing efficiency when allowing for frame-based identification, B is treated the same as N, i.e., a rectangular log-mel energy spectrum picture. In addition, the CNN kernel extracts local features from the log-mel power spectrum picture, enabling researchers to examine local time and frequency patterns **(Sehgal & Kehtarnavaz, 2018)**.

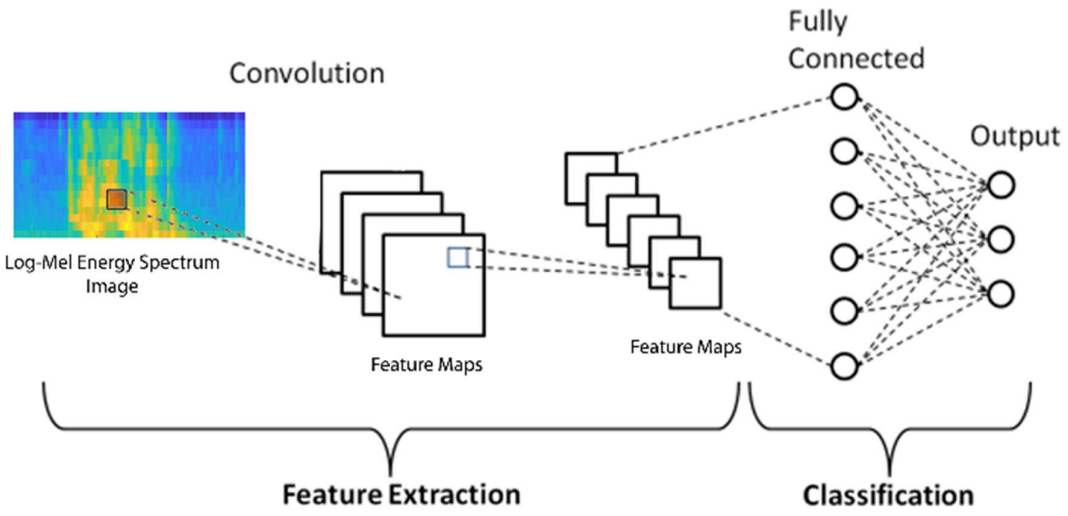

Figure 3. CNN Illustrations Are Inputted by Log-Mel Energy Spectrum Images

The objective function, which is also the ReLU objective function, is often used to achieve computational performance. It is shown in Equation (4).

$$\text{ReLU}(x) = \max(0,x) \tag{4}$$

The initialization layer for ReLU has a size value of 0 when x is negative, while x is the value of the input function, as well as the output, is the same as the input when x is positive.

2.3 HiVAD Audio Processing

The best parameter for HiVAD is a reference voltage of 16 kHz with a collection sample rate of 400 tests, or 25 milliseconds, and a 50 percentage variance. Due to a mismatch between the HiVAD context subtraction parameter and the lowest latency i/o feature, all events must be synchronized. As seen in Figure 4, HiVAD conducts this configuration on a trying to frame basis that maintains the shortest time delay.

A voice from microphones is transmitted at a duty cycle of 64 samples/sec, with a sample rate of 48 kilohertz. As described in **(Michaeltyson, 2017)**, spherical buffers are used to capture sampling frequency up to the necessary 600 measurements or 12.5 milliseconds of overlap, which is equivalent to the overlapping. The higher frequencies above 8 kHz will be filtered out, and frames are lowered via a bandlimited lowpass filter. After that, time is extracted by selecting three samples from the bandlimited sample per three seconds. The result a frame of 200 models at 16 kHz and a period of 12.5 ms. Overlapping frames are combined with previous overlapping edges to form a 25 ms processing frame or 400 models. The explanation for this is that MFSC is derived between 300 Hz and 8 kHz, which is where the majority of the talk frequency material is found.

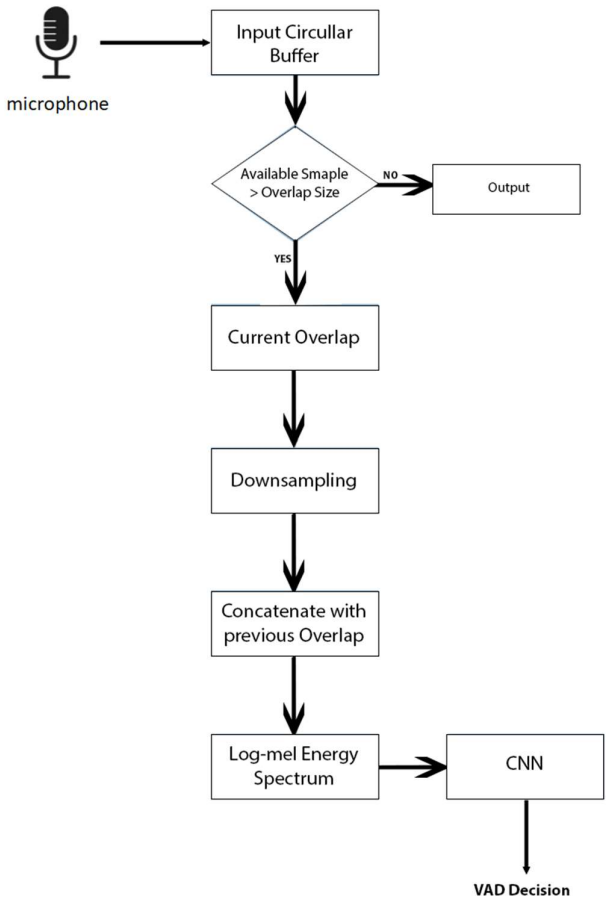

Figure 4. Flowchart of The HiVAD

2.4 Software Tools

The CNN HiVAD algorithm implemented at MATLAB includes input image formation and labeling. Tensorflow software with Python is used to perform CNN learning offline using input files. Tensorflow seems to have a C++ API that can be used to execute just a portion of CNN on smartphones. The trained offline-trained CNN is turned into a special inference framework that can be put to use in real-time operations or monitoring on mobile networks by removing the reverse propagation, planning, and detachment layers.

Using the device shell built-in **(Kehtarnavaz, et al, 2020)**, The CNN-based VAD's image production or extracting features module is encoded in C to create mobile applications. The application programs for Android smartphone applications is developed in Java, and the Superpowered APK device kit completes sound input/output. **(Superpowered, 2019)**

2.5 Low Latency

Reading and writing file voice sampled with a bit rate of 64 samples or 1.34 md and a sampling frequency of 48 kHz is needed to implement the minimum surface audio on Android smartphones. With audio optimization techniques, VAD is programmed to work at its best while maintaining these latency limitations. On Android smartphones, the size of the i/o frame varies depending on the manufacturer. On the Android Google Pixel, for reference, the lowest data rate with the interventions is 192 tests or four milliseconds at 48 kHz.

2.6 Training CNN

The input images must be collected scene by scene to run real-time VAD applications. However, it is not necessary to identify each frame individually. So as a consequence, a multiple threads method should be used for grouping. The image is created as the primary audio input/output chain, and CNN runs on synchronous parallel threads. This frees up compute space in the direct audio input/output chain for unit operations in the speech recognition pipeline.

Adam's optimization algorithm **(Kingma & Jimmy, 2014)** is used to practice the Model Convolution Neural Network, with the bridge as a disadvantage. Cross-entropy losses are determined as follows for classification model activities. It is shown in Equation (5).

$$loss = -(y * \log(p)) + (1 - y) * \log(1 - p) \quad (5)$$

Where y seems to be the CNN output that represents the likelihood of voice + sound, with 0 for the sound frame and One again for voice + sound frame, while p seems to be the CNN output that expresses the chance of speech + sound, with 0 for sound only frame and one again for voice + sound frame.

Models are conditioned over a period of 12 epochs, each of 12800 iterations. The learning standard was gradually lowered for the first six training days, beginning at 10-3, then at 10-4 for the following four periods, and finally at 10-5 for the final two periods. For teaching, The researchers used a ten-fold quasi pass scheme, with one crease used for testing as well as the rest for training.

2.7 Graphic User Interface

Figure 5 depicts the Android application's user interface. The Graphical User Interface (GUI) includes buttons for beginning and halting programs and displays of CNN classification performance, regular buffer view, and frame processing time.

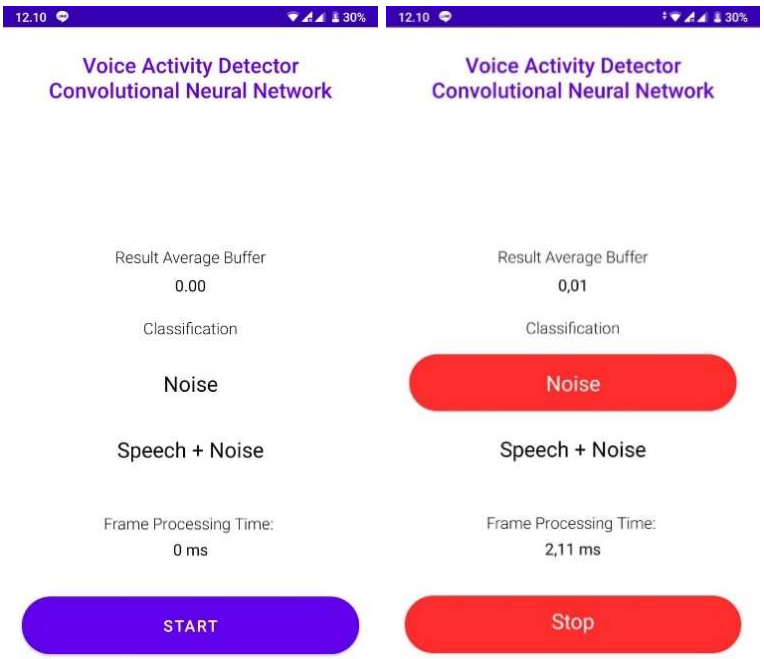

Figure 5. The User Experience of The Program

3. RESULT AND DISCUSION

FSDD datasets are used to train and test CNN VAD **(Zohar, et al, 2018).** The dataset consists of 3000 audio files from 6 speakers. In addition, the DCASE 2017 challenge dataset **(Mesaros, et al, 2017)**, which contains 15 distinct background noise conditions, was used as the noise dataset. For assessment, all speech sentences are used.

Subband characteristics are extracted for RF VAD, and log-mel imagery energy is extracted for CNN HiVAD. Both classifiers are tested using a tenfold bridge scheme. A sampling frequency of 16 kilohertz was used, the image was calculated for a 25 ms sample size with 50% overlap. The low volume is 300 Hz, the high rate is 8 kHz, the feature map is 40, and the FFT width is 512 bins for such log-mel energy range. Images of the log-mel energy continuum are collected approximately per 62.5 msec, as well as the likelihood contribution from CNN HiVAD is based on the current and past pictures. With eight subbands and an FFT size of 512, subband features are extracted. VAD decision-making efficiency must be stabilized, a median fine-tuning filter is added to approximately 20 frames.

The same data set is used to test VAD G729B and Sohn. The G729B utilizes the script from **(Mathworks, 2017)**, although Sohn utilizes the VAD script from **(Brookes, 2019).**

To assess HiVAD, we use Speech Hit Rate (SHR), the fraction of actual speech frames correctly classified as speech, and Noise Hit Rate (NHR), which is the fraction of noise frames correctly classified as noise. For most speech processing systems, it is imperative to get both SHR and NHR high because a lower NHR and SHR mean inaccurate assessments of noise and speech frames, respectively.

Tables 1 and 2 show the difference in mean inaccuracy depending on the Noise Hit Rate (NHR) and Speech Hit Rate (SHR) from previous studies. Judging from this table, HiVAD outperformed the percentage of other VAD methods with test results indicated by an increase in the average accuracy of NHR of around 43.91% compared to SOHN, 1.95% compared to RF, and 152.67% compared to G.729B. Then the increase in the average SHR accuracy of HiVAD was about 28.98% compared to Sohn, 42.13% compared to RF, and 15.89% compared to G.729B at the 0 dB SNR level.

Table 1. NHR Average

G729B	SOHN	RF	HiVAD
39,3	69	97,4	99,3

Table 2. SHR Average

db	G729B	SOHN	RF	HiVAD
0	81,8	73,5	66,7	94,8
5	85,4	79,8	78,9	92,8
10	88,8	83,9	85,6	90,0

To train and evaluate the HiVAD that has been developed, clean speech and noise were added at three different Signal-to-Noise Ratio (SNR) levels of 0, +5, and +10 dB to generate noisy speech. The lower the SNR level, the higher the noise. The higher the SNR level, the higher the sound signal.

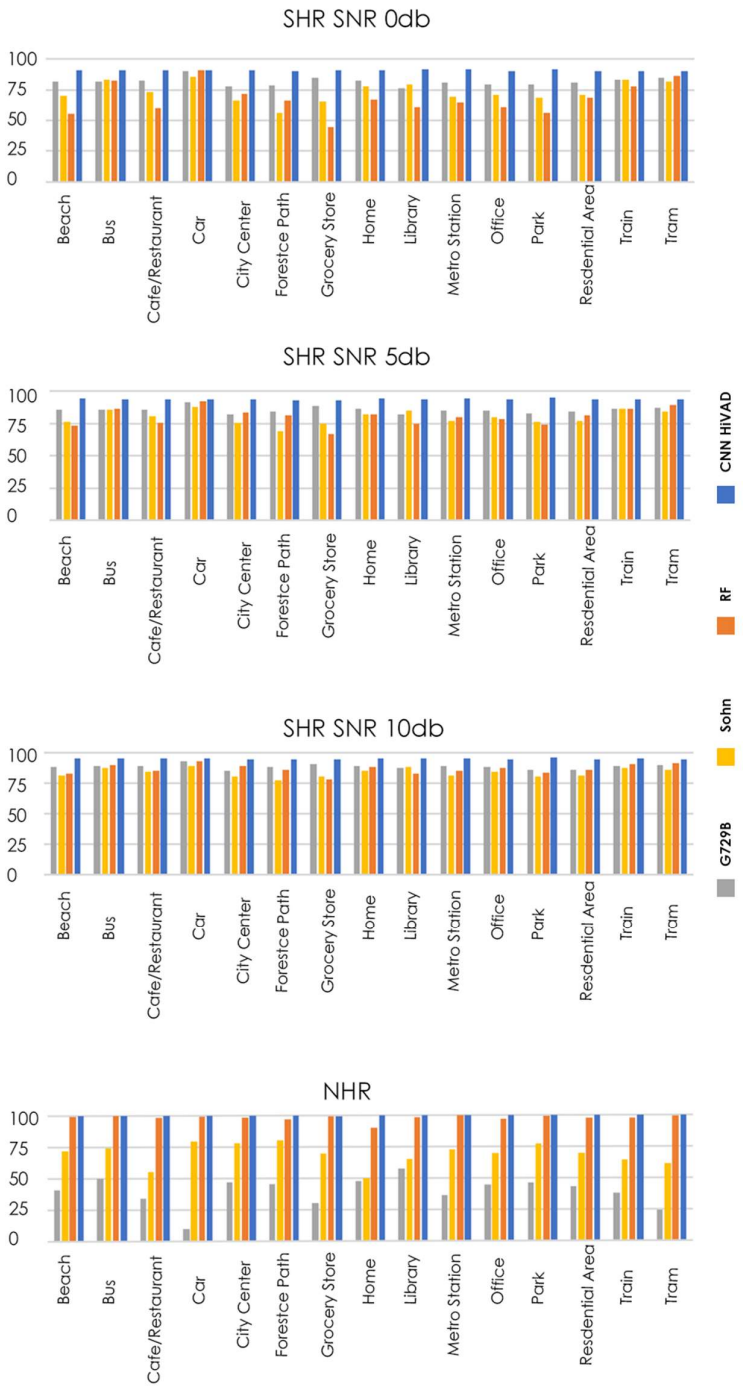

Figure 6. SHR And NHR of Four VAD in Different Noise Atmospheres

CNN HiVAD produces high SHR and NHR at every decibel (dB) while RF VAD, Sohn, and G729B are lower for each noise environment. There are 15 noises tested by the datasets trained, including beach, bus, café/restaurant, car, city center, forest path, grocery store, home, library, metro station, office, park, residential area, train and tram, the 15 neighborhoods were tested and based on Figures 6 CNN HiVAD shows high SNR and NHR. The RF VAD generated

a low SHR for 0 dB SNR and the statistical VADs generated low NHRs and inflated SHRs. A ratio higher than 1:1; greater than 0 dB; indicates more signal than noise.

4. CONCLUSION

HiVAD can track sound activity in real-time with a 64-sample or 1.34-millisecond audio delay buffer. HiVAD uses a convolutional neural network architecture to process audio frames in real-time, ensuring that no edges are missed and that good sound behavior detection accuracy is maintained. Multi-threading is a computation time-efficient mechanism for the real-time execution of other signal processing systems. It is used to render programs run concurrently with the primary audio line. The results revealed that convolutional neural network-based applications outperformed random forest-based applications.

REFERENCES

Brookes, M. (2019). *VOICEBOX: Speech Processing Toolbox for MATLAB*. Retrieved from http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html

Chandra, A. (2018). *Voice Activity Detection Sederhana Menggunakan Python*. Retrieved from https://medium.com/warung-pintar/membuat-voice-activity-detection-menggunakan-python-d13763ea277f#

Dong, E., Liu, G., Zhou, Y., & Zhang, X. (2002). Applying support vector machines to voice activity detection. *International Conference on Signal Processing Proceedings, ICSP*, (pp. 1124–1127).

Jo, Q. H., Chang, J. H., Shin, J. W., & Kim, N. S. (2009). Statistical model-based voice activity detection using support vector machine. *IET Signal Processing*, *3*(3), 205–210.

Kehtarnavaz, N., Sehgal, A., Parris, S., & Azarang, A. (2020). Smartphone-based real-time digital signal processing: Third edition. In *Synthesis Lectures on Signal Processing* (Vol. 11, Issue 2).

Kingma, D. P., & Jimmy, B. (2014). *Adam: A Method for Stochastic Optimization*. Retrieved from https://arxiv.org/abs/1412.6980

Krizhevsky, A., Sutskever, I., & E. Hinton, G. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, *60*(6), 1–1432.

Mathworks. (2017). *G.729 Voice Activity Detection—MATLAB & Simulink*. Retrieved from https://www.mathworks.com/help/dsp/examples/g-729-voice-activity-detection.html

Mesaros, Annamaria, Heittola, Toni, & Virtanen, T. (2017). *TUT Acoustic scenes 2017*. Zenodo. Retrieved from https://zenodo.org/record/400515#.YI0uhbUzbIU

Michaeltyson. (2017). *TPCircularBuffer*. Retrieved from https://github.com/michaeltyson/TPCircularBuffer

Obuchi. (2016). Framewise speech-nonspeech classification by neural networks for voice activity detection with statistical noise suppression. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2016*, (pp. 5715–5719).

Ramírez, J., Yélamos, P., Górriz, J. M., Segura, J. C., & García, L. (2006). Speech/non-speech discrimination combining advanced feature extraction and SVM learning. *International Conference on Spoken Language Processing, INTERSPEECH 2006 - ICSLP*, (pp. 1662–1665).

Rishi, S. (2019). *Audio Classification Using CNN — An Experiment*. Retrieved from https://medium.com/x8-the-ai-community/audio-classification-using-cnn-coding-example-f9cbd272269e

Saki, F., & Kehtarnavaz, N. (2016). Automatic switching between noise classification and speech enhancement for hearing aid devices. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, *2016-Octob*, (pp. 736–739).

Sehgal, A., & Kehtarnavaz, N. (2018). A Convolutional Neural Network Smartphone App for Real-Time Voice Activity Detection. *IEEE Access*, *6*, 9017–9026.

Superpowered. (2019). *Superpowered. Android Audio SDK, Low Latency, Cross Platform, Free*. Retrieved from https://superpowered.com/

Thad, H., & Keir, M. (2013). Recurrent neural networks for voice activity detection. *EEE International Conference on Acoustics, Speech and Signal Processing 2013*, (pp. 7378–7382).

Thomas, S., Sriram, G., George, S., & Hagen, S. (2014). Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (pp. 2538–2542).

Yang, X., Tan, B., Ding, J., Zhang, J., & Gong, J. (2010). Comparative study on voice activity detection algorithm. *Proceedings - International Conference on Electrical and Control Engineering, ICECE 2010*, (pp. 599–602).

Zhang, X. L., & Wu, J. (2013). Deep belief networks based voice activity detection. *IEEE Transactions on Audio, Speech and Language Processing*, *21*(4), 697–710.

Zohar, J., César, S., Jason, F., Yuxin, P., Hereman, N., & Adhish, T. (2018). *Free Spoken Digit Dataset (FSDD)*. Retrieved from https://www.kaggle.com/joserzapata/free-spoken-digit-dataset-fsdd.