Evaluating the Effect of Multiple Filters in Automatic Language Identification without Lexical Knowledge

Guan-Lip Soon[1]
Hardware Building Block (HWBB)
Motorola Solution (Malaysia) Sdn Bhd
Penang, Malaysia

Nur-Hana Samsudin[2]
School of Computer Sciences
Universiti Sains Malaysia,
Penang, Malaysia

Dennis Lim[3]
PCR Software Architecture Group
Motorola Solution (Malaysia) Sdn Bhd
Penang, Malaysia

***Abstract*—The classical language identification architecture would require a collection of languages independent text and speech information for training by the system before it can identify the languages correctly. This paper also address language identification framework however with data has been downsized considerably from general language identification architecture. The system goal is to identify the type of language being spoken to the system based on a series of trained speech with sound file features and without any language text data or lexical knowledge of the spoken language. The system is also expected to be able to be deployed in mobile platform in future. This paper is specifically about measuring the performance optimisation of audio filters on a CNN model integration for the language identification system. There are several metric to gauge the performance identification system for a classification problem. Precision, recall and F1 Scores is presented for the performance evaluation with different combination of filters together with CNN model as the framework of the language identification system. The goal is not to get the best filter for noise, instead to identify the filter that is a good fit to develop language model with environmental noise for a robust language identification system. Our experiments manage to identify the best combination of filters to increase the accuracy of language identification using short speech. This resulting us to modify our pre-processing phase in the overall language identification system.**

***Keywords*—Language identification; speech recognition; speech filters; minimal language data; minimal lexical information; optimal performance**

I. Introduction

Spoken language recognition system is the process of automatically determines the identity of a language spoken by a speech audio signal. The research on speech recognition has been started since 1930s progressing in the fifth generation of speech recognition starting from early 2000s to beyond 2020s [1]. Currently, the fifth generation focuses on key areas of improving recognition reliability, detecting and correcting linguistic irregularities and system robustness in detecting the speech against noise with the extension of machine learning model [1] that aligns to the objective of this paper.

Spoken language recognition is more challenging than text based language recognition because to date there is no single system that can offer a 100% error free recognition [2]. If we are to look at the human auditory perception towards a language, there are two categories identified in Zhao et al. [3]:
(1) Pre-lexical information where human able to distinguish an unfamiliar language, and
(2) Lexical semantic knowledge where human able to understand the semantic reasoning behind the words spoken.

A pre-lexical information can be referred to as how words are pronounced in individual terms, also known as phones, which also being applied in both speech recognition and text-to-speech system [4]. There are 6,909 languages in the whole world [5] with 200 to 300 different phones able to represent all available (recorded) languages [6].We conducted the analysis of this paper with the assumption that all languages which is considered in the language identification system will have overlaps pre-lexical information, such as phones sounds, but at the same time they are unique enough to be distinguished from one language to another with the information from non-overlaps information as well. This idea is also supported by Muthusamy et al. [7] and Zissman [8] where their studies on human test subject confirm that pre-lexical information can be used to classify language. Experiments have also been carried out to ascertain the pre-lexical information able to perform language identification with lower error rate on human test subject [9], [10].

The challenge of this project as a whole is to employ automatic language identification (LID) through acoustic level feature accurately and to identify or classify the languages on an embedded device. The targeted device has also a set of predefined language settings which make the implementation of this idea is fitted into the device criteria. This paper focuses on evaluating the effect of multiple filters in order to find the most optimal filter to be used in the subsequent processes: feature extraction and post-processing towards a robust language identification system.

The acoustic level feature will be extracted through Mel-Frequency Cepstral Coefficient (MFCC) with filter-banks represent the cepstral coefficients across time. The feature will be further transform into frequency domain for better vector representation of each coefficient. The extracted speech feature will be further conditioned through cepstral mean and variance normalization (CMVN) to normalize the extracted phoneme acoustic feature, where background noise is expected. Basic frequency warping is applied to normalize the speech of difference voices that will improve accent and regional intonation differences. The conditioned features will be trained

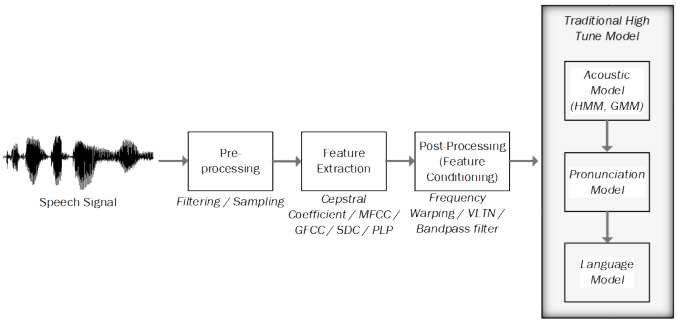

Fig. 1. A Conventional Speech Recognition System Pipeline Described by Schalkwyk [11].

on a Convolution Neural Networks (CNN) to classify the language. The evaluation metric will use standard classification evaluation metric.

Our intention of the implementation is also to break-away from conventional speech recognition system. Around 2014 researchers began using neural network to develop speech recognition system [11]. The conventional building block of speech recognition system comprises of an acoustic model that extrapolate audio speech to phonemes, a pronunciation model that connects the phonemes to form lexical words, and a language model that can be used to express semantically correct sentence [11]. Our motivation of cutting down the conventional layers of speech recognition system with the aid of neural network is to improve latency and processing requirement.

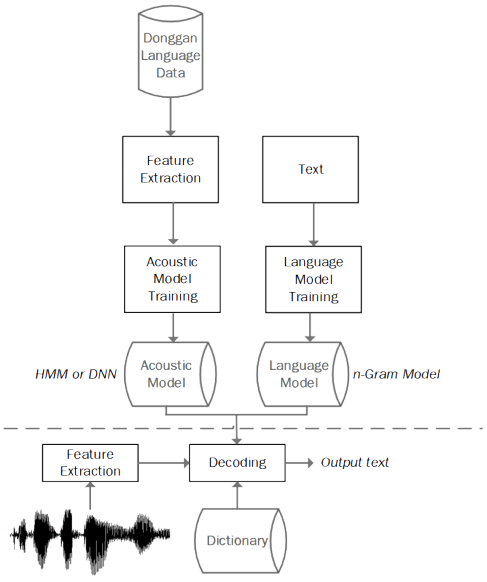

Fig. 2. A Conventional Speech Recognition System by Xu et al. [12].

Fig. 1 showed the traditional speech recognition system comprises of Acoustic Model, Pronunciation Model and Language Model. This is quite similar to Fig. 2 which is a general architecture of speech recognition system for Donggan province's language. This architecture is slightly different than the conventional speech recognition system without Deep Learning Model where it comprises an Acoustic Model and Language Model that utilizes statistical model of GMM and HMM. Another relatable architecture is shown in Fig. 3 that uses machine learning model classifier through SVM and decision tree.

With the goal to simplify our language identification system, we reduce the implementation footprint based on these related architectures to construct a speech language identification system using neural network variation model.

This paper is organised into five parts. In Section II, we will give some overviews of relevant work in language identification in speech recognition system. This will be followed by Section III where we will highlight some relevant techniques related to the methodology implemented by others to build their language identification system. Section IV describes the proposed solution to carry out based on the literature, system architecture review and assessment techniques based on relevant works presented. To determine the applicability and performance of the different filters suggested, Section V will first show a complete pipeline of our language identification. Section VI finally shows the effect of filters to the performance of our language identification system by discussing how the evaluation are being carried out and followed by the effect of the different filters for preliminary language identification. We end our paper in Section VII with the improvement in affect of this study towards the modification on the filters combination towards a better language identification system.

II. Background Studies on Language Identification Framework

This section will give an overview of relevant work in language identification in speech recognition system.

Duong and Duong [13] described acoustic audio technique to extract audio feature for voice pattern design. In this paper, the authors performed statistical model to conduct pattern recognition by using GMM, HMM and non-negative matrix factorization (NMF) on the extracted audio features. The main focus of Duong and Duong [13] as a relevant work was due to the input audio conditioning stage. There were two stages. During the first stage: pre-processing stage, Duong and Duong [13] extracted speech signal relevant frequency and frequency domain representation through Fourier Transform either using FFT, DCT or wavelet transform, all with filter. During the second stage, feature extraction using several relevant techniques like MFCC, Spectral Energy Peak (SEP) and Spectral Band Energy(SBE), Spectral Flatness Measure (SFM) and Spectral Centroid was observed. MFCC was the most preferred choice.

Mori et al. [15]'s paper was on spoken language identification closely related to the proposal of this paper which is a system without lexical knowledge. Their system able to detect six different languages. They were English, French, German, Dutch, Italian and Portuguese. All were from dataset derives from VoxForge. The feature extraction employed the MFCC alone and fed through a list of learning model and a statistical model. From the result of their paper, Neural Network (NN) performs better than Support Vector Machine (SVM) and GMM statistical model. Hence, this paper served as an important lesson for the endeavor of our project and justified that NN is best suited for the application of this project.

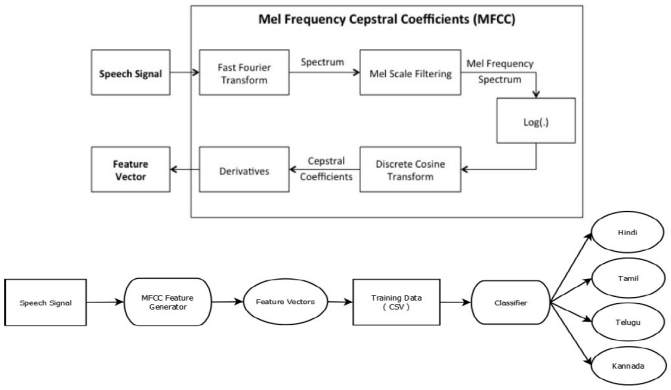

Fig. 3. An Example of Language Identification System Approach in Venkatesan et al. [14]. The Upper Portion show the Process of MFCC Extraction and the Bottom Portion shown how the Training Characterised the Speech Features into their Respective Languages

.

Vatin [16] worked on a language identification speech recognition system based on GMM. He discussed his effort during pre-conditioning of audio signal by using of filter to remove noise. Signal-to-Noise-Ratio (SNR) was introduced to measure the quality of desired audio signal against noise. In the case of our research, signal to noise ratio will highly dependent on device capability which is not the main objective. Some other technique employs for example Voice Activity Detector (VAD) to remove silence where users have a moment of silence before speech begins were also described in Vatin [16]. Other signal conditioning technique like RASTA filters were described to reduce vocal tract fluctuation from different speaker of the same language. The language identification system logic designed through fusing different GMM models was the focal research of his dissertation which is not the main focus of our research but will be very helpful to those developing non-dependent mobile device for language identification.

Van Segbroeck et al. [17] was an article on spoken language identification. The article is focused on computational cost consideration and recognition time requirement. The article showed the extracted audio feature had an impact on recognition performance. The identification of languages were modeled using GMM statistical model with each combined together as one large general speech models known as Universal Background Model. Van Segbroeck et al. [17] work concluded MFCC as the second fastest feature extraction method compared to FuSS (Fused Speech Stream) features. However, their approach required a very deliberate process where the utterance information would need to be transform into a low dimensional i-vector and then language classification methods will be applied. Should the focus of this research is not specific to the embedded device, FuSS could be a good alternative to explore into.

Aarti and Kopparapu [18] showed a language identification system that able to identify nine different Indian languages: Assamese, Bengali, Gujarati, Hindi, Marathi, Kannada, Malayalam, Tamil and Telugu through the use of Neural Network. A detailed high level diagram showed how the system comprised the process from pre-processing to feature extraction. There were using the common MFCC technique to perform feature extraction that suggested the language detection is without lexical knowledge on the speech data.

Venkatesan et al. [14] showed a language identification system that able to identify 4 types of Indian language: Tamil, Kannada, Telugu, and Malayalam through the use of Machine learning like Decision Tree and Support Vector Machine (refer to Fig. 3). The result of Accuracy for SVM was around 76% and Decision Tree was around 73% with feature extraction which employed MFCC technique without lexical knowledge on the speech data.

In summary of the relevant works suggested MFCC and neural network may provide a better yielding detection performance of language with quick recognition time.

III. Review on Methodologies used in Language Identification Systems

This section highlighted some relevant techniques related to the methodology implemented by others to build their language identification system. This section provide a review on the proposed methods and also highlighting the component's focus for this paper and our future work.

A. Pre-Processing

This stage is where the input audio signal will be digitized for processing. The actual implementation of this project will be deployed on STMicroelectronics STM32 Nucleo-64 MCU Development Board NUCLEO-F446RE. In this pre-processing stage, every speech audio that is needed to be trained in the proposed project requires to achieve a certain level of speech quality consistency. The stages in pre-processing stage can be further breakdown on the following:

1) Sampling rate or sampling frequency: Sampling rate represent the time domain sample rate of an audio signal. The higher the sampling rates the better the signal representation of a the speech's sinusoid signal. However, it also means the larger the storage space required. The minimum sampling rate is dictated by Nyquist theorem, where a sample signal frequency denoted by $f_{sample} = 2f_{source}$ which means the frequency of the sample must at least be twice of the signal frequency source [19].But in speech recognition, quality recognition is a main focus, and there could be restriction of real time deployment for Voice over Internet Protocol (VoIP) where the sampling rate for VoIP usage is usually set to 16 kHz for decent speech quality [1].

2) Filter: Speech filter is functioned to extract speech signal by cutting off unnecessary polluting signal that is deemed undesirable such as noise. We can classify filters into four types: Low pass filter, bandpass filter, high pass filter and notch filter. For speech filtering, bandpass filter and low pass filter will be used where human audible range from 20Hz to 20kHz. The ranges of human generated phones can start from 31.5Hz to 16kHz range based on [1].

3) Hamming Window: Hamming window is a hypothetical signal frame used to cut or segment a short series of speech signal from the whole signal frame. It also provide to smooth discontinuities in speech signal [15], [1].

4) Time domain to frequency domain transformation: Speech signal is mostly represented using the time domain representation. However, in order to capture the most speech features in a speech recording, Discrete Fourier Transforrm (DFT) is used to represent the same time domain signal into frequency domain [20]. The conversion can be calculated through Fast Fourier Transform (FFT) [1] since FFT known to only consume lower computation resources during processing.

Another method known as Discrete Cosine Transform (DCT) offers an advantage of data compression capacity without capturing the full periodic amplitude of a speech signal. However, it will not be useful in multi-speech audio signal to segment of different speakers. Furthermore DCT compression will also omit non-linearity aspect of speech that usually present in real life [21]. Thus, DFT is able to reproduce the amplitude of the original speech signal. The expression of DCT is given in Vatin [16].

In summary between DFT, FFT, and DCT; DFT will be used during training stage of the dataset to reproduce the best possible representation of frequency domain to truly capture the speech signal as there are plans in future for the ability to detect main speaker's voice from background's speaker speech through amplitude or strength of the speech signal. FFT will be use for embedded device's end after prototype. Our pre-processing implement the stated pre-processing stage.

B. Feature Extraction

Feature extraction is the process of finding feature representation from the speech audio for language identification. The focus techniques are primarily on acoustic phonemes feature extraction. A few important technique developed is highlighted.

1) Mel-Frequency Cepstral Coefficients (MFCC): MFCC is a widely used speech recognition feature extraction method. It exploits auditory principles, where filter banks perform mapping of auditory response [2]. The amount of filter banks reflect the coefficients in terms directly correlate to the power spectral envelop for each frame [16], [18]. In speech recognition system, there will be speaker to speaker variation audio imprinting accuracy feature which must not to be too accurate in feature extraction. Else, it will lead to miss-classification. Another reason it is widely used is due to the fact that speaker dependency can be greatly reduced in speech processing system [14].

MFCC can be referred to as Mel-Frequency Filter Bank (MFFB) without Discrete Fourier Transform (DCT). The use of DCT in learning model is due to highly correlated input data requires décor-relation usually require in machine learning model that does not perform well in highly correlated data. Without DCT only deep learning model able to handle highly correlated data present in the characteristic of speech signal [21].

2) Perceptual Linear Prediction Coefficients (PLP): is also based on frequency domain spectral plot. In Hermansky [22], it is introduced as a mean to offer computation efficiency to extract audio feature. It has one disadvantage due to susceptibility towards noise. The cepstral coefficients to reflect the acoustic phonemes are computed out by autoregressive coefficient unlike MFCC designated by the filter banks set. Based on Van Segbroeck et al. [17], PLP is slower than MFCC.

3) Gammatone Frequency Cepstral Coefficients (GFCC): is based on auditory periphery model [23]. It uses Gammatone filterbank to generate a time sequence of Gammatone frequency. The Gammatone cepstral analysis is close to MFCC technique, where GFCC is transform into cepstrum through DCT [24]. GFCC shows greater detail of accuracy for speaker identification compared to MFCC. However, greater detail in accuracy means limitation of performance according to Van Segbroeck et al. [17].

4) Mean-Hilbert Envelope Coefficients (MHEC): is an extension of GFCC offers better noise immunity and distortion over MFCC according to Sadjadi and Hansen [25]. The recognition error rate has an average of 1.5% better than MFCC as observed on the result according to Sadjadi and Hansen [25].

5) Fused Speech Stream (FSS): FSS is a technique to combined different feature extractions technique to yield better recognition time known as feature fusion through Principle Component Analysis (PCA) according to Van Segbroeck et al. [17]. The feature will be trained on the extracting feature techniques. For fused speech stream, every input speech signal will have to go through the techniques before language model processing. Thus, implementation of this technique has to be carefully considered in computing restrictive environment. It may benefit during model training stage but not during implementation stage of model processing.

MFCC offers the fastest recognition time in terms of processing among the audio feature extraction techniques with descent feature integrity. Also, as explain in Fayek [21], without DCT in MFCC, only deep learning model able to handle highly correlated data using only the characteristic of speech signal.

C. Post-Processing or Feature Conditioning

Post-processing is an augmentation technique to improve existing feature extraction method against noise, and the variation of speech. Feature conditioning must not be over-applied as it will impact the integrity of the feature that will compromise the accuracy in language identification. This paper will not be covering the post-processing stage. However, it is suffice to highlight CMVN, RASTA and VLTN are applied after feature extraction stage to condition and improve extracted features colluded by noise in our implementation.

D. Model Stage

The modelling of our language identification system will be based on Convolution Neural Network (CNN) using the feature extracted from MFCC. CNN is chosen because it treats the frequency domain as an array image. CNN offers better discriminating effect on interfering noise if presents in the image array, which is not present in RNN where it is in time-series domain. Another reason CNN is more favourable in speech recognition over Deep Neural Network (DNN) is because DNN input is represented as vector resulting losing structural locality and susceptible in echo interference [26] as compared to CNN.

E. Noise Cancelling and Noise Filtering Algorithm

Signal conditioning through filtering and noise cancellation are important to provide speech audio quality in an actual environment with actual noise colluded. The best quality on speech signal is always through the use of noise cancellation. Noise cancellation techniques usually require reference noise signal to provide close to silence noise cancellation effect. This type of noise cancellation is sometimes referred as Active Noise Cancellation (ANC) [27]. Thus, hardware implementation will be needing external mic source which can be difficult to implement in a software-only-implementation that are limited on the proposed language identification system.

Thus, Digital Signal Processing (DSP) filter will be used as a means of filtering noise both applies in training stage, testing stage and deployment stage. SNR or signal-to-noise ratio will be used as judging criteria of the filter.

Filtering is another way of noise filtration process with the ability of dampening noise but not total elimination of noise. There are two types of filters: Linear filter and non-linear filters. Linear filters will be used to chop-off and extract out specified band of frequencies to ease the Fourier transformation of the signal in subsequent feature extraction without considering the whole audio spectrum. Linear filters also help in certain none-speech related noise that can be filtered off to improve the human audible signal. However, the linear filters has a limitation in terms of audio signal which can be seen as poor filtering in non-additive noise like impulse noise or "salt and pepper" noise induced by the recording devices (hardware's internal circuitry switching) noise or sampling noise [28], [29]. Therefore, non-linear filters are also used to provide a solution against non-additive noise.

The four standards linear filters are usually used in signal processing based on the following [30], [31]:

- Butterworth Filter offers maximally flat response in passband with slower roll-off frequency during cut-off, can be improved by increasing poles or orders of the filter at the expense of computation power.
- Chebyshev Filter offers sharper roll-off than Butterworth filter at the same poles or orders therefore much efficient in transition band from passband to stopband. The sharpness of roll-off rate is placed second behind the Elliptic filter. One major disadvantage is high ripple response in pass band.
- Bessel/Thompson Filter has the flattest response but with poor roll-off rate compared to Butterworth filter and with slower magnitude response compared to Butterworth filter. It is much suitable for high-bandwidth signal filtering with flat response
- Elliptic Filter has the steepest roll-off at the same poles and orders compared to Chebyshev Filter but with both pass-band and stop-band with large ripples compared to Chebyshev filter

Hence, Butterworth filter will be selected for as the first stage filtering. A 6 poles configurations were chosen due to prevalent chosen number in the industry. The number of poles used dictates the implementation cost for both hardware and computational cost. However, the higher the poles (number) offers sharper (and thus better) roll-off response to the target cut-off frequency.

Median filter works by replacing the median value of the neighbouring signal. The filtering window size is determined by the number of neighbours. Median filter does not work well with large filter size as it will cause the signal to lose its integrity resulting under-sampling effect in auditory quality which the sound will appear robotic or contain aliasing. A highly noisy signal will not be effective to be filtered by Median filter due to the filter reliance on neighbouring sampling quantized points [29].

Wiener filter is an algorithm gearing towards the ability to reduce noise with a reference noise signal to reconstruct back the signal as good as possible. It is possible to employ a randomized reference noise signal with designated power level in miliwatt region for example 0.2mW is good against impulse for non-additive noise especially in single channel microphone approach. Wiener Filter works by sensing noise present and estimate the amount of reference noise added to feedback to the system and reduce it. This approach is limited to non-additive noise and thus it is not a very efficient noise reduction algorithm.

IV. Proposed Solution

This section describes the proposed solution carried out based on the literature, system architecture review and assessment techniques and results based on relevant works presented. The focus of our study is to find a solution for a language identification system via speech interface with the best and direct approach and minimal language dependent lexical information so that it is feasible to be applied in an embedded device. As a start, we proposed the flow as shown in Fig. 4 which has a single pipeline model.

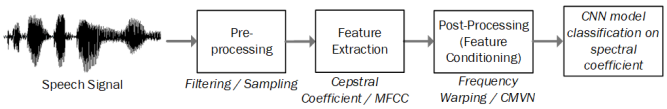

Fig. 4. The Initial Proposed Language Identification System.

The proposed language identification system architecture in Fig. 4 served as a guide towards a complete language identification system. In this paper, we are going to conduct exploratory analysis on the best possible pre-processing stage, specifically in filtering the speech signal in order to achieve the best performance system. The exploratory data analysis and evaluation is actually involved in all stages: pre-processing stage, feature extraction and post-processing stage to serve the same purpose.

Our pre-processing analysis consists of evaluating different filtering techniques. The combination of filters are to be tested by alternating and combining a selected filter or a series of filters which include: pre-emphasis signal conditioning, Butterworth High Pass Filter, Butterworth Band Pass Filter, Median Filter, Wiener Filter and three stage filters (combination of three filters: Butterworth, Median and Wiener filters). There are 10 permutations all together.

The following are the steps for our pre-processing stage:

1) Speech signal is framed at 25ms size..
2) The spectrogram representative in 10ms stride is prepared for the next phase (feature extraction).
3) nFFT=512, Full used would be 22050 where nFFT=552 for full spectrum based on Sampling Time/FilterBanks [32].
4) Filter Bank is set at 40. Too low of a filter banks will not capture the phoneme features from speech signal.
5) Apply permutation filters based on below:
    a) Pre-emphasis conditioning
    b) Butterworth Bandpass 275Hz to 7kHz
    c) Median Filter
    d) Wiener Filter

In short, the audio signal is captured in 3GP format as default codec. Then, it will be reconverted to MP3 before feeding to the feature extraction module in the standard PC. The feature extraction will further convert in raw *FLAC* format to perform further processing. It is suffice to highlight that the 3GP format is a standard issue of the device input where direct conversion to *FLAC* file is not recommended due to limited capacity of the device. The *FLAC* format was chosen over wave file format due to it size's capacity as well as lossless audio quality. Comparing *.wav* superior lossless quality and the concerns of memory size, *.flac* file was chosen due to its compressed lossless feature [33].

The different sets of filters are introduced due to their different filtering behaviour. Pre-emphasis filter is functioned to increase the frequency magnitude (within a frequency band) the magnitude of some (usually higher) frequencies with respect to the magnitude of other (usually lower) frequencies in order to improve the overall speech signal. Butterworth bandpass and highpass filter response with cut-off audible low frequency noise from 1Hz to 275Hz, and 7kHz and beyond to prevent down-modulated high frequency components. The Median filter is for removing sampling noises and white noise effectively, while Wiener offers noise reduction that can be used to remove or dampened noise in audible human speech range.

V. Implementation

In order to show the effect of filters to the performance of our language identification system, a complete pipeline is needed. The following describes the process how the implementation is carried out.

This project is a classification problem in identifying the type of language based on the 'file name assigned' with the extracted cepstral coefficient that correlates the phoneme features of the specific spoken language in *FLAC* file format.

Although the focus is to minimise the lexical data, some information are required to classify the trained data to the correct language group. Thus, a proper naming convention and some info is required for all training and validation files.
(language)_(gender)_(recording_ID).fragment(index)
[.(transformation)(index)] .flac
is the representative wave file name.

The attributes comprise gender type and speaker ID. Each mp3 files is around 45 minutes long that can be broken into 10 seconds instances with differences in spoken speed, pitch and noise, with 29 files each for German, English and Spanish. The cepstral feature will be varied in speed from 0.8 to 1.2 ratio with 1 as normal speed and pitch level varied from -200 to 200 ratio. The variation on the mp3 audio files will cater for different speeds and pitch level spoken by a variety of users.

The conditioning of the audio files starts by introducing noise files converted into FLAC format, normalize with a standard 16 bit and sampling rate mono before 10 second splits applies to all training files and test file as shown in Fig. 5. The original mp3 files, normalized files before split, and converted FLAC files before normalizations are removed to conserve space of the machine. The script will be carried out as task behind the task manager alleviating in speed up the CPU processes in the background. This phase still CPU intensive since no GPU intervention can be used when extracting and preparing the audio files.

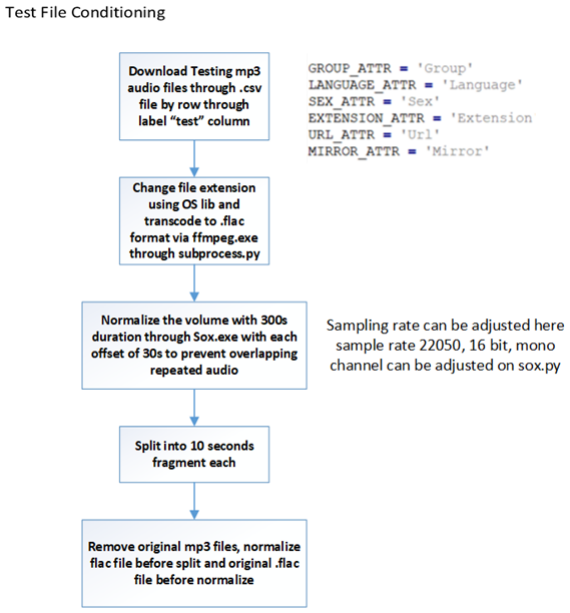

Fig. 5. The Steps for the Test Files Segmentation in every 10 Seconds Range

The expected outcome of this project is an application level prototype, with the embedded device user interface able to change its interfacing languages by speaking to the device. The expectation of the classifier performance for the initial prototype will be based on learning based related works as a benchmark [15] with an accuracy performance of 91% classifying between German and Italian using Neural Network alone, while [18] classifying 9 different Indian languages with accuracy performance around at most 44% with deep neural network (DNN).

The two prior work show a trend with a single model approach multi-class classification with large sets of language may impact overall performance. The maximum languages for the expected prototype will support four languages and minimum two languages to meet a highly accurate performance: $>80\%$.

For this paper's implementation there are three languages for the filter evaluation model: German, English and Spanish. The methodology used for this project will focus primarily on

the optimization of pre-processing filters to improve the overall classification abilities of the deep learning model

VI. Evaluation and Results

For fundamental evaluation on speech recognition system, it can be evaluated based on accuracy on the amount of accurately classified language. But, to reveal more of a system performance, miss detection and false alarms can be used [34].

F1 metric is also a widely used metric for evaluating classification or decision making system [34]. More frequently use metric, like precision, computes the fraction of correctly labeled positive samples on the input and recall, is the fraction of correctly labeled samples among the positive samples [34]. In this paper, we will evaluate the accuracy of our language identification system by evaluating the performance using different version of filtering (and some combination) before the language identification process by this metric.

$$F_1 = \frac{2TP}{FN+FP+2TP} \; or \; F_1 = \frac{2\times precision\times recall}{precision+recall}$$
$$Recall = \frac{TP}{TP+FN}$$
$$Precision = \frac{TP}{TP+FP}$$

This research look for the performance of different filters permutations. The following section will be based on German, English and Spanish dataset as a fixed constant with different filters permutation. The dataset will be based on 67,860 entries with 5,220 validation entries, and 540 test entries. These are the representation of 180 test set for each language.

A. Validation Set Performance

Using 5,220 validation entries, different filters permutation are implemented to observe the best model performance of default topology. The best feature extraction line up will be selected for further model optimization. Table I showed the precision, recall and F1 score from individual language in validation set.

There are 10 different permutation of filters' configuration applied in this observation as shown in Table I. No filter refers to the default filtering. Pre-emphasis refers to pre-emphasis for input frequency range most susceptible to noise. Butterworth refers to Butterworth Bandpass 275Hz (and lower) to 7kHz(and higher) to filter. Median and Wiener both are referring to Median filter and Wiener filter respectively. Median filter can remove sampling noise and white noise effectively while Wiener filter offers noise reduction that can be used to remove or dampened noise in audible human speech range. Default Wiener size is 3 but the observation is also carried out with different size with the assumption the algorithm can learn better fir on slightly broader spectrogram bandwidth visibility. The filters that is coming after that is a combination of a few filters mentioned above. 3 stage filters is referring to a combination of three mentioned filters: Butterworth filter, Median filter and Wiener filter size 5.

The different in size is necessary because when we combined Median filter and Wiener noise cancellation filter, the filter size of these two methods must not be the same because the filter magnitude is used to dampened the noise and thus compromising the auditory quality. To prevent such situation, size 5 is selected for Wiener filter and size 3 is used for Median filter. The odd number sizing is used due to the Median filter averaging effect. The filter size on each averaging operation of the Median filter will degrade in terms of quantization and sampling rate of the quality of the signal to dampen off noise. Hence, filter size of Median filter is a cost trade off between signal quality and noise reduction. Thus, the higher the filter size, the lower the audio signal quality. This resulting Median filter size 5 cannot be used. Thus, Median filter of size 3 is still being used but for the Wiener filter, a higher filter size will need to be used to attain significant noise dampening effect after the Median filter is applied. Size 5 is minimal size used for Wiener because filter increment must be an odd number for convolutioning noise effect on the spectrogram.

The code of languages accronym are as follows: de is Deutsche or German Language, en is English (mix) and sp is Spanish.

TABLE I. Precision, Recall and F1 Score on the Validation Set.

Feature Permutation	Validation Set = 5,220 entries								
	Precision			Recall			F1 Score		
	de	en	sp	de	en	sp	de	en	sp
No filter	.94	.98	.92	.95	.9	.99	.94	.94	.95
Pre-emphasis	.96	.96	.93	.95	.94	.96	.95	.95	.94
Butterworth	.95	.97	.94	.95	.94	.97	.95	.95	.95
Median	.96	.94	.95	.94	.96	.94	.95	.95	.94
Wiener size 5	.96	.97	.94	.96	.94	.97	.96	.96	.95
Wiener size 3	.94	.96	.94	.96	.93	.96	.95	.95	.95
Butterworth + Median	.95	.97	.94	.95	.95	.96	.95	.96	.95
Butterworth + Wiener 5	.96	.97	.93	.95	.94	.97	.95	.95	.95
Butterworth + Wiener default	.94	.96	.95	.95	.95	.96	.95	.95	.95
3 stage filters	.96	.98	.95	.95	.97	.97	.96	.98	.96

In summary, we can conclude the standard metric results on validation set to be as shown in Table II. Number of iteration column is referring to the value obtain after a routine check for recognizing overfitting or underfitting of data. Although the longer a network (in this case CNN) is trained, the better it performs on the training set, at some point, the network fits too well to the training data and loses its capability to generalize [35].

TABLE II. Summary of Standard Metric Results on Validation Set

Feature of Permutation	Standard Metric			
	Accuracy	No of Iteration	Average Confidence	Average Precision Score
No Filter	0.9454	20	0.9699779	0.93
Pre-emphasis	0.9477	14	0.959	0.92
Butterworth	0.9498	24	0.9627879	0.96
Median	0.947893	14	0.95913273	0.9
Wiener size 5	0.954	24	0.96213144	0.93
Wiener size 3	0.949234	22	0.95767	0.95
Butterworth + Median	0.9544	14	0.9646244	0.94
Butterworth + Wiener 5	0.95287	20	0.961755	0.97
Butterworth + Wiener default	0.9521	19	0.96329778	0.96329778
3 stage filters	0.962835	23	0.97275305	0.98

TABLE III. STATISTICAL HYPOTHESES GUIDELINES TO READ THE BINARIZED CONFUSION MATRIX. THE GRAYED AREA IS THE VALUES LISTED IN TABLE IV FOR DIFFERENT FILTERS AND LANGUAGES.

Reality	Study Finding	
	Negative	Postive
Negative	True Negative	False Positive
Positive	False Negative	True Positive

B. Test Set Performance

This section will provide the evaluation obtained from test set for the three languages. Table IV shows the binarized confusion matrix for each languages. To aid in reading the binarized confusion matrix table in Table IV, Table III will be the guide for reading Table IV. The language acronyms are similar to the validation set. In totally, each language will have 180 speech recordings test of the evaluated language and 360 other recordings belonging to the other two languages.

Table V shows the comparison of different filters confusion matrix on 540 entries test set. These are extracted from language specific binarized confusion matrix as shown in Table IV. From the confusion matrix, 3 stage filters yield the best performance. Individually, Deutsche entries classified with 7 false negatives with 0 false positives. English entries have 1 false negative and 4 false positives. Spanish entries have 4 false positives and 0 false negatives. However, Butterworth bandpass filter with Wiener size 5 alone yield better performance in Spanish language entries with 2 false positives and 0 false negatives.

From Table V the best performance on Confusion Matrix is still three stage filtering system comprises Butterworth Bandpass filter starting from 275Hz to 7 kHz, Median Filter and Wiener Filter. Among 180 test entries for German Language, there are four entries classified as English and three entries classified as Spanish. As for English, among 180 test entries, 1 is classified as Spanish. Spanish is perfectly classified for the 180 test entries.

The binarized confusion matrix in Table IV and the combinational confusion matrix in Table V can be summarised as Table VI. It shows the best performance is the three stage filters combination for each language for precision, recall and F1 score on the test set. However, Median filter performs worst in test set where the precision is less than 0.9 for English and recall is less than 0.9 for Deutsche.

VII. CONCLUSION

From the evaluation in the previous section, we have shown the best performance for filtering is the three stage filters combination where for each language's precision, recall and F1 score on the test, the precision, recall and F1 score are consistently good as compared to other filters or combination filters. Thus, based on this evaluation, the three stage filters produces a better language identification system.

Based on our results, a minor modification on our pre-processing stage can improve the overall language identification accuracy. Thus, Fig. 4 is best to be modified into Fig. 6.

Based on the modified pre-processing, the research can further proceed into the next stages: feature extraction, post-processing towards the completion of the CNN model. In feature extraction, the fastest recognition time in terms of processing among the audio feature extraction techniques with descent feature integrity is looked for. Post-processing will complement the extracted features to improve existing signal feature extraction method against noise, and the variation of

TABLE IV. BINARIZED CONFUSION MATRIX ON INDIVIDUAL LANGUAGE TEST SET

de		en		sp	
360	0	344	16	354	6
17	163	4	174	1	179

No filter

de		en		sp	
359	1	340	20	356	4
22	158	3	177	0	180

Default pre-emphasis

de		en		sp	
358	2	353	7	357	3
8	172	3	177	1	179

Butterworth bandpass filter

de		en		sp	
355	5	336	24	357	3
26	154	2	178	4	176

Median filter

de		en		sp	
357	3	350	10	357	3
10	170	3	177	3	177

Wiener filter size 3

de		en		sp	
357	3	343	17	357	3
18	162	2	178	3	177

Wiener filter size 5

de		en		sp	
356	4	346	14	357	3
14	166	6	174	1	179

Butterworth + Median filter

de		en		sp	
359	1	350	10	357	3
9	171	4	176	1	179

Butterworth + Wiener default filter

de		en		sp	
360	0	353	7	358	2
8	172	1	179	0	180

Butterworth + Wiener filter size 5

de		en		sp	
360	0	356	4	356	4
7	173	1	179	0	180

3 stage filters

TABLE V. COMPARISON OF DIFFERENT FILTERS FOR COMBINATIONAL CONFUSION MATRIX ON 540 ENTRIES TEST SET FROM THE BINARIZED CONFUSION MATRIX FOR INDIVIDUAL LANGUAGES.

	de	en	sp
de	163	15	2
en	0	176	4
sp	0	1	179

No filter

	de	en	sp
de	158	20	2
en	1	177	2
sp	0	0	180

Pre-emphasis

	de	en	sp
de	172	7	1
en	1	177	2
sp	1	0	179

Butterworth

	de	en	sp
de	154	24	2
en	1	178	1
sp	4	0	176

Median

	de	en	sp
de	162	17	1
en	0	178	2
sp	3	0	177

Wiener size 5

	de	en	sp
de	170	9	1
en	1	177	2
sp	2	1	177

Wiener default

	de	en	sp
de	166	13	1
en	4	174	2
sp	0	1	179

Butterworth + Median

	de	en	sp
de	171	9	0
en	1	176	3
sp	0	1	179

Butterworth + Wiener default

	de	en	sp
de	172	7	1
en	0	179	1
sp	0	0	180

Butterworth + Wiener 5

	de	en	sp
de	173	4	3
en	0	179	1
sp	0	0	180

3 Stage Filters

TABLE VI. PRECISION, RECALL AND F1 SCORE FOR TEST SET AFTER EXTRACTING FROM CONFUSION MATRIX.

Feature Permutations	Test Set = 540 entries								
	Precision			Recall			F1 Score		
	de	en	sp	de	en	sp	de	en	sp
No filter	1	.92	.97	.91	.98	.99	.95	.95	.98
Pre-emphsis	.99	.9	.98	.88	.98	1	.93	.94	.99
Butterworth	.99	.96	.98	.96	.98	.99	.97	.97	.97
Median	.97	.88	.98	.86	.99	.98	.91	.93	.98
Wiener (size 5)	.98	.91	.98	.9	.99	.98	.94	.95	.98
Wiener (default)	.98	.95	.98	.94	.95	.98	.96	.96	.98
Butterworth + Median	.98	.93	.98	.92	.97	.99	.95	.95	.99
Butterworth + Wiener 5	1	.96	.99	.96	.99	1	.98	.98	.99
Butterworth + Wiener 3	.99	.95	.98	.95	.98	.99	.97	.96	.99
3 stages filters	1	.98	.98	.96	.99	1	.98	.99	.99

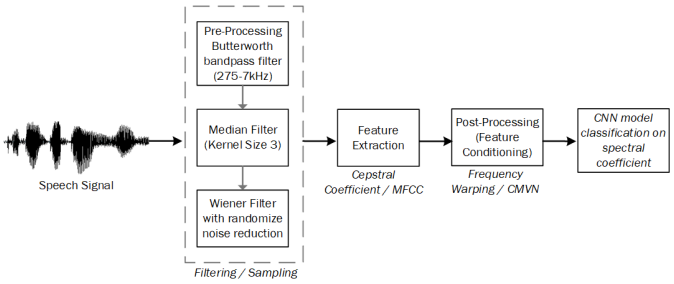

Fig. 6. Overall Modified Filtration within the Whole Architecture.

speech. It is hoped by having all the fined tune process will improve the language identification accuracy for a system without lexical knowledge.

ACKNOWLEDGMENT

The implementation on the dedicated device are copyrighted of ©2020 Motorola Solutions, Inc. All Rights Reserved. The authors would like to thank the insight of all experts - DSP techniques: Ondy Sukma, system level integration: Pang W. K, Sim Yew Tatt & Ondy Sukma, and Java related platform programming: Tan Chun Mun as well as special embedded level advice: Leong Kim Hong. This paper publication is funded by the Universiti Sains Malaysia's Short term grant no: 304/PKOMP/6315273.

REFERENCES

[1] J. Benesty, M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*, ser. Springer Handbook of Speech Processing. Springer Berlin Heidelberg, 2007. [Online]. Available: https://books.google.com.my/books?id=P-g3DwAAQBAJ

[2] H. Li, B. Ma, and K. A. Lee, "Spoken language recognition: From fundamentals to practice," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1136–1159, May 2013.

[3] J. Zhao, H. Shu, L. Zhang, X. Wang, Q. Gong, and P. Li, "Cortical competition during language discrimination," *NeuroImage*, vol. 43, no. 3, pp. 624–633, nov 2008. [Online]. Available: https://doi.org/10.1016%2Fj.neuroimage.2008.07.025

[4] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 1st ed. USA: Prentice Hall PTR, 2000.

[5] M. P. Lewis, *Ethnologue: Languages of the World*, 16th ed. Texas: SIL International.

[6] M. Ashby and J. Maidment, *Introducing Phonetic Science*. Cambridge University Press, mar 2005. [Online]. Available: https://doi.org/10.1017%2Fcbo9780511808852

[7] Y. K. Muthusamy, E. Barnard, and R. A. Cole, "Reviewing automatic language identification," *IEEE Signal Processing Magazine*, vol. 11, no. 4, pp. 33–41, 1994.

[8] M. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, p. 31, jan 1996. [Online]. Available: https://doi.org/10.1109%2Ftsa.1996.481450

[9] Y. K. Muthusamy, N. Jain, and R. A. Cole, "Perceptual benchmarks for automatic language identification," in *Proceedings of ICASSP '94. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. i, 1994, pp. I/333–I/336 vol.1.

[10] D. Van Leeuwen, M. Boer, and R. Orr, "A human benchmark for the nist language recognition evaluation 2005," January 2008.

[11] J. Schalkwyk, "An All-Neural On-Device Speech Recognizer," Internet, Google AI, 2019, accesed on March 2020. [Online]. Available: https://ai.googleblog.com/2019/03/an-all-neural-on-device-speech.html

[12] H. Xu, H. Yang, and Y. You, "Donggan speech recognition based on deep neural network," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, 2019, pp. 354–358.

[13] N. Q. K. Duong and H. Duong, "A review of audio features and statistical models exploited for voice pattern design," *CoRR*, vol. abs/1502.06811, 2015. [Online]. Available: http://arxiv.org/abs/1502.06811

[14] H. Venkatesan, T. V. Venkatasubramanian, and J. Sangeetha, "Automatic language identification using machine learning techniques," in *2018 3rd International Conference on Communication and Electronics Systems (ICCES)*, 2018, pp. 583–588.

[15] J. D. Mori, M. Faizullah-Khan, C. Holt, and S. Pruisken, "Spoken language classification," 2012. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.278.7614&rep=rep1&type=pdf

[16] C. Vatin, "Automatic spoken language identification," Master's thesis, 2012.

[17] M. Van Segbroeck, R. Travadi, and S. S. Narayanan, "Rapid language identification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 7, pp. 1118–1129, 2015.

[18] B. Aarti and S. K. Kopparapu, "Spoken indian language classification using artificial neural network — an experimental study," in *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2017, pp. 424–430.

[19] E. Ayanoglu, "Data transmission when the sampling frequency exceeds the nyquist rate," *IEEE Communications Letters*, vol. 1, no. 6, pp. 157–159, 1997.

[20] E. W. Weisstein, *Fourier Transform*, 2020 (accessed July 30, 2020), http://mathworld.wolfram.com/FourierTransform.html.

[21] H. Fayek, *Speech Processing for Machine Learning: Filter banks, Mel-Frequency Cepstral Coefficients (MFCCs) and What's In-Between*, April 2016, https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html.

[22] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, apr 1990. [Online]. Available: https://doi.org/10.1121%2F1.399423

[23] R. D. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice, "An efficient auditory filterbank based on the gammatone function," in *IOC Speech Group on Auditory Modelling at RSRE*, vol. 2, no. 7, 1987.

[24] Y. Shao, S. Srinivasan, and D. Wang, "Incorporating auditory feature uncertainties in robust speaker identification," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, vol. 4, 2007, pp. IV–277–IV–280.

[25] S. O. Sadjadi and J. H. Hansen, "Mean hilbert envelope coefficients (MHEC) for robust speaker and language identification," *Speech Communication*, vol. 72, pp. 138–148, sep 2015. [Online]. Available: https://doi.org/10.1016\%2Fj.specom.2015.04.005

[26] S. Park, Y. Jeong, and H. S. Kim, "Multiresolution cnn for reverberant speech recognition," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*, 2017, pp. 1–4.

[27] Say-Wei Foo, T. N. Senthilkumar, and C. Averty, "Active noise cancellation headset," in *2005 IEEE International Symposium on Circuits and Systems*, 2005, pp. 268–271 Vol. 1.

[28] G. George, R. M. Oommen, S. Shelly, S. S. Philipose, and A. M. Varghese, "A survey on various median filtering techniques for removal

of impulse noise from digital image," in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, 2018, pp. 235–238.

[29] S. Vishaga and S. L. Das, "A survey on switching median filters for impulse noise removal," in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, 2015, pp. 1–6.

[30] H. Zumbahlen and A. D. Inc., *Linear Circuit Design Handbook*. USA: Newnes, 2008.

[31] Y. Sun, "ELE314 Linear Systems and Signals - Classic Filters," Internet, The University of Rhode Island, 2018, accesed on March 2020. [Online]. Available: https://www.ele.uri.edu/courses/ele314/handouts/YS06_Classicfilters.pdf

[32] S. Adam, "Plotting & Cleaning - Deep Learning for Audio Classification p.3," Youtube, 2018, accessed on March 2020. [Online]. Available: https://youtu.be/mUXkj1BKYk0

[33] W. Gordon, "What's the Difference Between All These Audio Formats, and Which One Should I Use?" Lifehacker.com, 2012, accessed on January 2020. [Online]. Available: https://lifehacker.com/what-s-the-difference-between-all-these-audio-formats-5927052

[34] D. Zhu, H. Li, B. Ma, and C. Lee, "Optimizing the performance of spoken language recognition with discriminative training," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1642–1653, 2008.

[35] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, jun 2018. [Online]. Available: https://doi.org/10.1007%2Fs13244-018-0639-9