JOIV : Int. J. Inform. Visualization, 8(1) - March 2024 279-288

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Performance Analysis of Feature Mel Frequency Cepstral Coefficient and Short Time Fourier Transform Input for Lie Detection using Convolutional Neural Network

Dewi Kusumawati [a], Amil Ahmad Ilham [b,*], Andani Achmad [a], Ingrid Nurtanio [b]

[a] *Department of Electrical Engineering, Hasanuddin of University, Bontomarannu, Gowa, 92171, Indonesia*
[b] *Department of Informatics, Hasanuddin of University, Bontomarannu, Gowa, 92171 Indonesia*
Corresponding author: [*]amil@unhas.ac.id

***Abstract*— This study aims to determine which model is more effective in detecting lies between models with Mel Frequency Cepstral Coefficient (MFCC) and Short Time Fourier Transform (STFT) processes using Convolutional Neural Network (CNN). MFCC and STFT processes are based on digital voice data from video recordings that have been given lie or truth information regarding certain situations. Data is then pre-processed and trained on CNN. The results of model performance evaluation with hyper-tuning parameters and random search implementation show that using MFCC as Voice data processing provides better performance with higher accuracy than using the STFT process. The best parameters from MFCC are obtained with filter convolutional=64, kerneconvolutional1=5, filterconvolutional2=112, kernel convolutional2=3, filter convolutional3=32, kernelconvolutional3 =5, dense1=96, optimizer=RMSProp, learning rate=0.001 which achieves an accuracy of 97.13%, with an AUC value of 0.97. Using the STFT, the best parameters are obtained with filter convolutional1=96, kernel convolutional1=5, convolutional2 filters=48, convolutional2 kernels=5, convolutional3 filters=96, convolutional3 kernels=5, dense1=128, Optimizer=Adaddelta, learning rate=0.001, which achieves an accuracy of 95.39% with an AUC value of 0.95. Prosodics are used to compare the performance of MFCC and STFT. The result is that prosodic has a low accuracy of 68%. The analysis shows that using MFCC as the process of sound extraction with the CNN model produces the best performance for cases of lie detection using audio. It can be optimized for further research by combining CNN architectural models such as ResNet, AlexNet, and other architectures to obtain new models and improve lie detection accuracy.**

***Keywords*— MFCC; STFT; CNN; detection; lies; parameters.**

Manuscript received 16 Aug. 2023; revised 12 Oct. 2023; accepted 25 Dec. 2023. Date of publication 30 Mar. 2024.

I. INTRODUCTION

Lie detection is a process to identify whether someone is lying or not. This is important in areas such as law, security, and psychology. Several backgrounds underlie the importance of lie detection, one of which is the need to maintain security, especially in Indonesia, where in recent years, crimes such as corruption and terrorism have been rampant. Hence, the need for accurate information is critical during the investigation process. Therefore, it is necessary to have a lie-detector model that can assist investigators in the investigation process. The importance of this research is that when someone lies, their speech pattern changes so that it can be detected through changes in pitch, loudness, and speed during speaking. These changes can be analyzed using various signal processing techniques; therefore, this research seeks the process of more effective voice processing to build a lie detection model through voice**.**

Various methods can detect lies, including language analysis body, EEG, P300, and polygraph tests. However, this method is based on previous research that resulted in accuracy that has not been maximized other than that it can be influenced by factors such as anxiety and individual habits. Hence, lie detection must be done carefully, considering the various factors affecting the detection results. The existing lie detector tools are based on previous research is a polygraph that measures the response of the nervous system [1], EEG which is a signal the brain uses to recognize information hidden in the brain to detect lies, but the EEG method has a broader application [2], [3]. Several methods and algorithms are used in research on lie detection in speech, including SVM models, Bayesian models (BN), conditional random field models (CRFM), DBN, CNN, LSTM, and RNN[4],

According to studies, the accuracy of signal classification into statements of truth or falsehoods utilizing 14 channels of EEG data as input to a convolution neural network is up to 84.44% [5]. Lie detection using P300 is used for the lie detection method, then the transformation method. To extract features from the previously processed electroencephalogram signal, a brief Fourier time was used [6]. Another thing demonstrates that the most frequently reported indexes for lie detection are the interviewee's physical traits, facial expressions, gaze direction, and bodily movements[7]. In terms of research [8] combines many modalities, which includes audio, video, EEG, and eye gaze, but in this study for audio using a KNN classifier with an accuracy of 56%. Research conducted by [9], where using pupil dilation for lie detection with the help of robots.

Deception Detection in Videos using the Facial Action Coding System, which employs audio analysis with features of Cepstral Coefficients (CC) and Spectral Regression Kernel Discriminant Analysis classification (SRKDA), is one of the researchers that have executed lie detection study using audio. The CC feature is used to represent voice signals in the frequency domain and time, where the resulting accuracy is 77.72% [10]. Deception Detection Research using Real-life Trial Data [11], this study uses audio tracks from video recordings trial as a voting modality. Next, the verbal features are extracted from the audio track using the unigram and bigram methods. Research conducted by[12] where the accuracy obtained from the study was between 51.24% to 59.50%, The Mel-Frequency Cepstral Coefficients (MFCC) are used for audio feature extraction in audio processing. After the audio features are extracted using MFCC classified using GMM (Gaussian Mixture Model). In research conducted by[13], in the study, Pitch and the Mel Frequency Cepstral Coefficient (MFCC) were used to extract voice characteristics, the accuracy results obtained are 88.23% for the detection of lies and 84.52% for honesty detection.

Several methods have been previously developed in the detection of lies, among other things; Polygraph test the most common lie detection method the method used is to use a polygraph test tool or polygraph test [5]. A polygraph test monitors a person's physiological changes while answering questions certain questions, such as changes in heart rate, respiration, and blood pressure. However, this method is often considered inaccurate and is still heavily criticized. Another method is to analyze someone's statement. This method involves an examination of body language and analysis of sentences, intonation, and words used by someone in a statement. However, this method still has limitations because can be influenced by a person's habits or everyday language. fMRI (Functional Magnetic Resonance Imaging), another method of lie detection being used developed is by using fMRI [14].

Facial expressions are very crucial for fraud detection[11]. outlines possible micro expressions. When variables as the intensity and alignment of the face are taken into thought, brief, involuntary movements are an indication of deception. [15]. Utilizing LabVIEW, one can identify expressions and faces which is an indicator of deception, and extracting based elements geometric [16], [17], By extracting facial movements, a lie detection learning model is trained to utilize the coding system as a parameter. Study it uses LSTM [10]. Using text patterns to informally identify expressions as lies or facts allows for accurate lie detection 93% for BERT[18]. With an accuracy of 64%, multimodal lie detection using video, audio, EEG, eye movement, random forest approach, and KNN classification yielded the best results[8].

There have been several lie detection models employed in studies, including using EEG signals with different processing techniques [5], [13], [19]–[22]. The number of blinks determines the detection lie using the HAAR Cascade method[23]. Utilizing biologic signals, the DNN method is utilized to identify lies[1]. Computer vision and machine learning are utilized for fraud detection[24].

By seeing several previous studies, this research detects lies through sound by using the MFCC, Prosodic, and STFT processes for extraction that originates from voice input. The results of each extraction process will be input into the CNN model. This is due to the growing development of digital voice technology, as well as the ability of Convolutional neural network (CNN) methods are examples of deep learning algorithms. to process voice data. This study aims to perform a performance comparison between the MFCC, Prosodic, and process STFT technology in lie detection using the CNN deep learning algorithm, which is still not widely studied.

Our proposed lie detection system uses audio with a feature extraction process from MFCC, Prosodic, and STFT. MFCC uses a cepstral coefficient of 20, with the CNN deep learning model adding optimizers, namely ADAM, Adadelta, and RMSProp. Using random search to find the best solution from the proposed model that was built, achieved better classification with a higher level of accuracy than several previous related studies.

Comparing the process of processing voice input in lie detection, a model that is more effective in detecting lies can be found and can provide a better understanding of the differences in the performance of sound technology with the MFCC, Prosodic, and STFT processes based on the resulting level of accuracy. Compared to other feature extraction methods, such as Prosodic and STFT, the use of MFCC features will improve the accuracy of the speech lie recognition model. In addition, this research can also contribute to making a model that performs better in detecting lies, especially using audio so that it can be implemented into systems that can help in various fields such as law, business, and politics.

II. Materials and Methods

A. The Basic Concept of Lying

Lie detection is a process whether to determine someone is lying or not based on the signals emitted when someone speaks or performs certain actions [25]. Previous studies have shown that the detection of Lies can be done using sound signal processing technologies such as frequency, formant, and pitch analysis[26].

One of the sound processing technologies that are widely used in research lie detection is the CNN[5]. CNN is a kind of algorithm of deep learning that can recognize complex patterns in images or data in other high dimensions. In the context of lie detection, CNNs can be trained to recognize sound patterns associated with lying or telling the truth.

Apart from that, there are also other sound processing techniques such as MFCC [1], [27], which can be used to analyze the sound properties associated with lying. MFCC is a calculated numerical representation of the sound signal and represents the sound energy spectrum in the form of coefficients ordered according to the level of relevance with a human voice.

B. MFCC and STFT

MFCC is one of the extraction techniques features used in speech signal processing, in the 1980s David first introduced this technique, and has been since the feature extraction technique that is most commonly used in speech signal processing. The technique is based on cepstral transformation, which converts the sound signal to the frequency domain from the time domain. The MFCC filter sakes the logarithm of the signal's power spectrum sound and applies a Mel bank filter on that power spectrum to generate the cepstral coefficient. This cepstral coefficient is then used as a feature for analysis of further information, such as classification or speech recognition. MFCC has been shown to be effective in various speech signal processing applications, including lie detection of signals sound [13].

The Fourier transform is a mathematical transformation that decomposes the function in the time domain into its constituent frequencies. This transformation is widely used, especially in signal processing (signal processing). STFT is the Fourier transform used to determine sinusoidal frequencies on the local part of the signal as it changes with time. Several previous studies have tested the capabilities of sound technology digital and spectrograms to detect emotions [28].

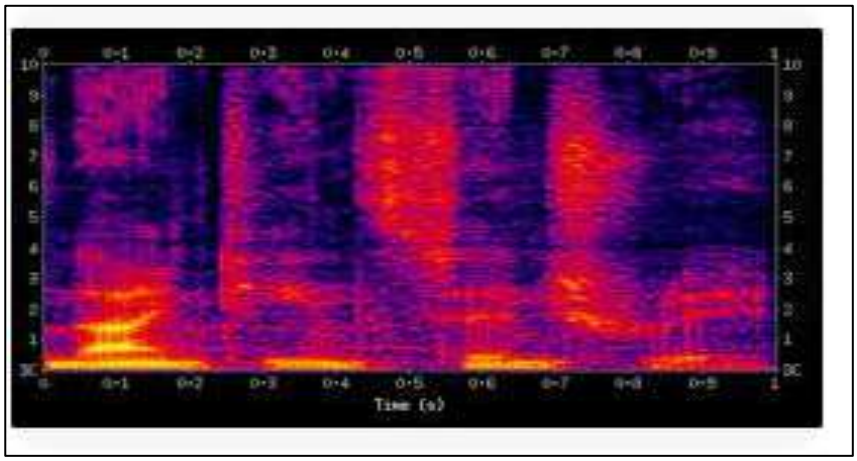

Fig. 1 Sample spectrogram images of audio

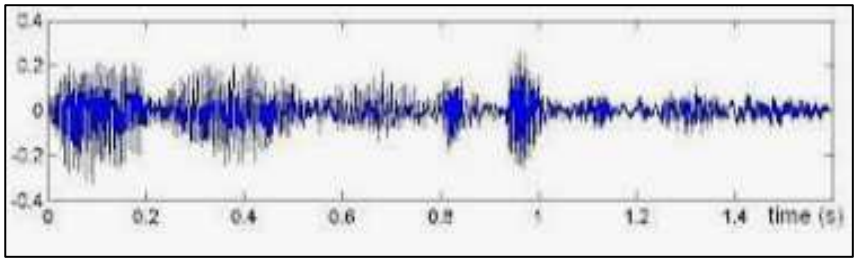

Fig. 2 Sample MFCC of audio

C. Optimizer

The optimizers used in this study are Adam, Rmsprop, and Adaddelta. Adam is one of the most used optimizers in the network of artificial nerves. Adam combines momentum algorithms and adaptive methods to set the training process's rate of learning. RMSprop (Root Mean Square Propagation) is a popular optimizer that adopts an adaptive method for setting the training process's rate of learning. This optimizer maintains the moving average of the squares of the previous gradient and uses that value to normalize the gradient during training. Adadelta is another optimizer as well using an adaptive method to set the learning rate. This optimizer combines its algorithm is similar to RMSprop, with the difference that it doesn't require a learning rate as an input parameter. Adadelta calculates the learning rate adaptively based on the moving average of the previous gradient[29].

D. Dropout

In deep learning, dropout is a regularization method used to avoid overfitting in the model. The multiple layers of a deep neural network allow the model to learn complex relationships between inputs and outputs. To prevent certain neurons from becoming overly dependent on other neurons and force the model to learn more general features, dropout removes a random portion of neurons at each training iteration. Dropping out a random portion of neurons can help the model avoid overfitting and improve generalization to data that has never been seen before [30], [31].

E. Prosidic

Prosodic feature extraction is a process to extract sound features related to intonation, rhythm, and stress in language. Prosodic features can be used to recognize emotion,

intonation, and meaning in language. Some of the commonly used prosodic features in speech recognition are tempo, duration, intonation, and stress. Prosodic features can be used in various applications, such as word recognition, speaker recognition, and emotion recognition. Prosodic characteristics are achieved by modulation of various acoustic features perceived by the listener. Prosodic features are understood in terms of fundamental frequency which is the basis for pitch (also intonation or melody), duration objectively measured as subjective length, intensity denoted as loudness, and spectral structure referred to as timbre[32].

F. CNN

Architecture of CNN consists of a convolutional layer, an activation layer, and a pooling layer. Several previous studies have used CNN in emotion detection based on sound analysis [29]. This architecture is usually used to process data in the form of images or images but can also be used for data with other structures such as voice or text. CNN consists of several layers with different functions, namely [23], 1) Convolution layer to extract features from images using performs a convolution operation between the filter and the input image. 2) Pooling Layer for performing down-sampling of the features that have been extracted by convolution layers. This aims to reduce the dimensions of the data and eliminate the features that are not important. 3) Activation layer to introduce non-linearity in CNN. Function a commonly used activation is the Rectified Linear Unit (ReLU), which generates a positive output if the input is positive and a zero output if the input is negative. 4) Fully Connected Layer to connect each node in the previous layer with each node in the next layer. This layer is usually found at the end of the CNN and is responsible for classifying input data into proper categories.

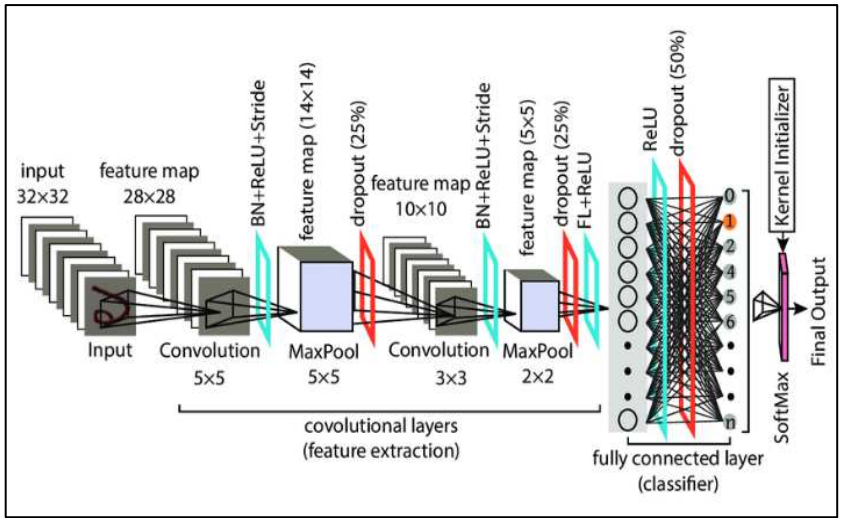

Fig. 3 CNN Architecture

G. Research Methodology

1) Data collection: The dataset used is sourced from a public dataset consisting of 121 videos. In this dataset, 61 videos were labeled lies and 60 videos were labeled truth [34][35]. Based on the guilty and not guilty verdicts, the video dataset has been labeled as lying or truthful. For the 28.0-second dataset, we performed an audio enhancement procedure. With the time-stretching approach, the audio duration was shortened from 28.0 to 4 seconds.

2) Data pre-processing: Audio signal trimming, normalization.

3) Feature Extraction

- The feature extraction procedure at MFCC is completed as follows: Pre-emphasis to increase speech signal clarity and reduce noise. Windowing, the audio signal is split into brief, closely spaced frames. A window (such as the Hamming window) multiplies each audio frame to lessen side effects. In this study, the frame is divided every 4 seconds. Each time frame is then calculated using the transformation power spectrum Fourier. The power spectrum is then converted to a Mel scale to suit the characteristics of human hearing. The cepstral coefficient represents the characteristics of the speech signal at each time frame.
- The feature extraction procedure at STFT is completed as follows:
  Pre-emphasis to increase voice signal clarity and reduce noise.Frame blocking. The sound signal is then divided into several time frames overlapping. Windowing. Each time frame is then applied by the windowing function to reduce the effect of discontinuities at the start and end of the frame. Fast Fourier Transform (FFT). Each time frame is then calculated as spectrum power using the Fourier transform.
- In Prosodic, the feature extraction process is as follows: Pre-emphasis to improve the clarity of the speech signal and reduce noise.Frame blocking. The speech signal is then divided into several overlapping time frames. Windowing. Each time frame is then applied with a windowing function to reduce the effect of discontinuities at the beginning and end of the frame.

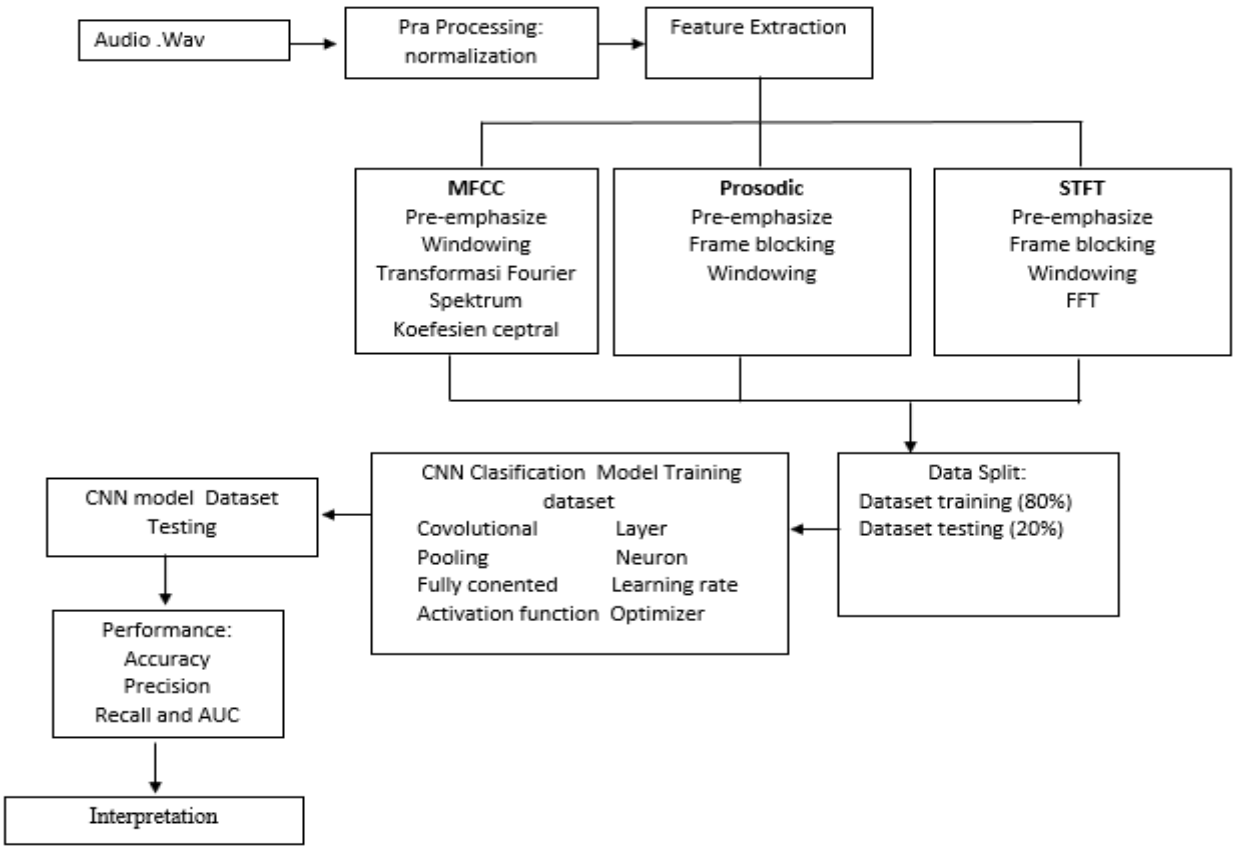

Fig. 4 Flow of Research Methodology

4) Split Data: After the respective feature extraction processes from MFCC, Prosodic, and STFT, the dataset which is used is divided randomly into two parts, namely the data training and data testing. The division of the dataset is carried out with a ratio of 80:20, where 80% of the dataset is used for model training and 20% is used for model testing. The division of the dataset is carried out after pre-processing and feature extraction is carried out on the dataset. After that, the dataset is divided into two parts using the function train_test_split () from the sci-kit-learn library in Python. Training data is used to train the CNN model, while data testing is used to test the performance of the model.

5) CNN Model Training: Each group from both MFCC and STFT was trained using a model CNN deep learning algorithm. The CNN model used will involve several convolutions, pooling, and fully connected layers for the classification process. Parameters such as the number of layers, the number of neurons, the learning rate, and optimization methods are set accordingly Experiment to get the best results.

6) CNN Model Testing: The model obtained during training is used for data testing to see the performance level of the CNN model in getting the best results.

7) Performance: Showing the performance of the CNN model using both the MFCC, Prosodic, and the STFT feature, of accuracy, Precision, Recall, and AUC

8) Interpretation and Conclusion: Results of analysis and comparison between digital sound groups using MFCC, Prosodic, and spectrogram groups with STFT features were evaluated and conclusions were drawn against the performance of each group in lay detection using the CNN model. Research results can contribute to the development of detection technology for more accurate and effective lies.

H. Pseudocode of MFCC, Prosodic, and STFT

1) The pseudocode of MFCC in this study is as follows:

Inputs:
Audio signals (waveforms)
Outputs:
MFCC features
Procedure:

- Read an audio signal data file.
- Set the maximum length.
- Calculate the MFCC formula with each parameter.
- Check the MFCC form.
- If the MFCC form is more than the maximum length, the length of the MFCC form is modified.
- And if the MFCC shape is less than the maximum length, do some padding techniques to make the shape equals.

End procedure.

2) The pseudocode of Prosodic in this study is as follows.

Inputs:
Audio signals (waveforms)
Outputs:
Prosodic features
Procedure:

- Read an audio signal data file.
- Extract prosodic feature.

- Save the extracted feature and labels.
- Initialize empty array to store feature and label.
- For each folder and label
- Get the list of audio files in folder.
- For each audio in the list
- Generate the full audio file path.
- Call the extract prosodic feature.

End procedure.

3) The Pseudocode of STFT in this study as follows:

Inputs:
An audio signal (waveform)
Outputs:
Spectrogram of STFT results
Procedure:

- Read an audio signal data file.
- Iterate through the audio signal => while frame_start + N <= len(x):
- Fetch frame => frame = x [frame_start:frame_start + N]
- Frame_windowed => Windowing (frame)
- Perform the Fourier Transform on the frame and then save the Fourier transform results.
- Updating the frame position

End procedure.

III. Results and Discussion

A. Data Description

The description of the data in this study includes the data used to train and test the CNN model as well as the results of processing and compiling the data that was carried out prior to model training. The data used in this study consisted of digital voice recordings recorded during the interview process and were categorized into two classes, namely lie and truth. The test results show that honesty and lying data have different characteristics in terms of nature and frequency distribution. Truthful data tends to have a lower frequency spectrum and lies tend to have a higher frequency spectrum.

The digital sound data is then converted into a spectrogram to clarify the frequency characteristics in the data. Furthermore, the spectrogram data is processed using data augmentation techniques to increase the variation of the training data. Data augmentation includes applying time shift, frequency shift, and amplitude shift. After data processing is complete, the data is divided into two parts, namely training data and test data. The training data is used to train the CNN model and the test data is used to test the model's performance. The performance of the CNN model is assessed based on a number of metrics such as accuracy, precision, recall, and f1-score. The results of the analysis show that the resulting CNN model can classify digital voice recordings as lies or honesty with high accuracy. In conclusion, it can be said that the right data processing and the use of the appropriate CNN model can produce optimal performance in lie detection based on digital voice recordings.

B. Results and Evaluation of The MFCC Model With CNN

In this study, we tried to use epochs from 100 to 1200 epochs.

TABLE I
PERFORMANCE MFCC - CNN

Model	Performance of MFCC - CNN					
	Classification	**Precision**	**Recall**	**F1-Score**	**Accuracy**	**Area Under Curve (AUC)**
1	Lie	0.96	0.97	0.97	96.63%	0.97
	Truth	0.97	0.96	0.96		
2	Lie	0.73	0.53	0.62	64.84%	0.66
	Truth	0.59	0.78	0.67		
3	Lie	0.96	0.98	0.97	96.51%	0.96
	Truth	0.97	0.95	0.96		
4	Lie	0.96	0.97	0.97	96.26%	0.96
	Truth	0.96	0.95	0.96		
5	**Lie**	**0.98**	**0.97**	**0.97**	**97.13%**	**0.97**
	Truth	**0.98**	**0.98**	**0.97**		

From Table I obtained best parameters using random search technique where the model that produces the maximum accuracy value is obtained from model 5 with an AUC value of 0.97 and an accuracy value of 97.13%. The best parameters obtained are as follows: filter convolutional1 = 64, kernel convolutional1 = 5, filter convolutional2 = 112, kernel convolutional2 = 3, filter convolutional3 = 32, kernel convolutional3 = 5, dense1 = 96, Optimizer = RMSProp, learning rate = 0.001, with an epoch value of 1.200. The test results show that the developed CNN model is able to recognize lies in digital voice data with an accuracy of 97.13%, with an AUC value of 0.97.

The test results also show that the developed CNN model has a high recall value for the "lying" class of 0.97 and a good precision value of 0.98. However, for the "no lie" class, the CNN model has a low precision value of 0.96 and an F1 score of 0.97.

C. Result and Evaluation of The STFT Model With CNN

From Table II the model that produces the maximum accuracy value is obtained from model 4 with an AUC value of 0.95 and an accuracy value of 95.39%. Best parameters obtained are as follows: filter convolutional1 = 96, kernel convolutional1 = 5, filter convolutional2 = 48, kernel convolutional2 = 5, filter convolutional3 = 96, kernel convolutional3 = 5, dense1 = 128, optimizer = Adaddelta, learning rate = 0.001 with an epoch value of 1000.

The test results also show that the developed CNN model has a high recall value for the "lying" class of 0.97 and a good precision value of 0.94. However, for the "no lie" class, the CNN model has a high precision value of 0.97 and a low recall

value of 0.93. From Table III the model that produces the maximum accuracy value is obtained from model 5 with an AUC value of 0.69 and an accuracy value of 68%. Best parameters obtained are as follows: filter convolutional1 = 64, kernel convolutional1 = 5, filter convolutional2 = 64, kernel convolutional2 = 5, filter convolutional3 = 128, kernel convolutional3 = 5, dense1 = 128, optimizer = Adam, learning rate = 0.0001 with an epoch value of 1000.

The test results also show that the developed CNN model has a high recall value for the "Lie" class of 0.76 and a good precision value of 0.80. However, for the "Truth" class, the CNN model has a high precision value of 0.79 and a low recall value of 0.79.

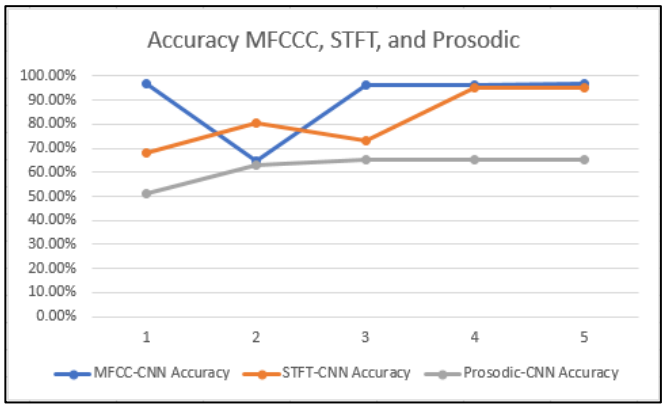

Fig. 5 Comparison Accuracy of MFCC, STFT and Prosodic

In Figure 5 shows that the maximum accuracy obtained from the MFCC-CNN is 97.13%, while for STFT-CNN an accuracy of 95.39% is obtained, and for Prosodic-CNN, an accuracy of 68% was obtained.

The AUC values of MFCC-CNN, STFT-CNN and Prosodic-CNN in each model can be seen in Figure.6. In Figure 6, the maximum AUC value obtained from the MFCC-CNN process is 0.97, for STFT-CNN an AUC of 0.95 is obtained, and prosodic CNN an AUC of 0.69 is obtained. In this study, we conducted various testing scenarios. First, by using epochs ranging from epoch 100 to epoch 1200. Second, the learning rate is 0.01 to 0.0001 and conducted several experiments, thus showing that the model has a fairly good generalization. In Tables I, II, and III, with the comparison of accuracy and AUC values in determining the classification of lie detection based on sound, there is no significant difference in each test result. In Tables I, II, and III it can be seen from the resulting value by adding a dropout layer to avoid overfitting.

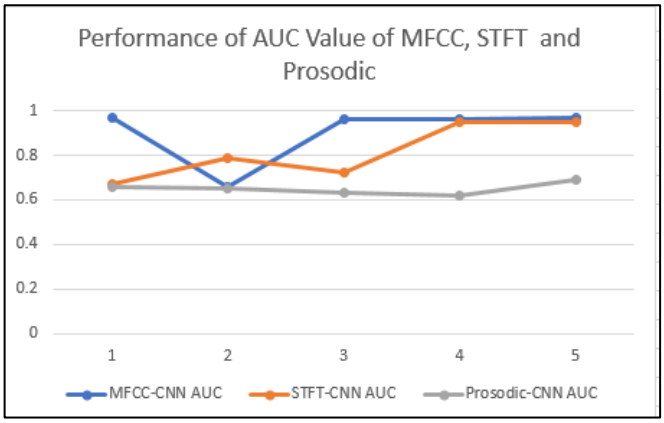

Fig. 6 Comparison AUC from MFCC-CNN, STFT-CNN and Prosodic-CNN

TABLE II
PERFORMANCE STFT - CNN

Model	Performance of STFT - CNN					
	Classification	**Precision**	**Recall**	**F1-Score**	**Accuracy**	**Area Under Curve (AUC)**
1	Lie	0.65	0.86	0.74	68.08%	0.67
	Truth	0.75	0.48	0.58		
2	Lie	0.74	0.96	0.84	80.42%	0.79
	Truth	0.94	0.62	0.75		
3	Lie	0.77	0.86	0.77	73.19%	0.72
	Truth	0.79	0.59	0.67		
4	**Lie**	**0.94**	**0.97**	**0.96**	**95.39%**	**0.95**
	Truth	**0.97**	**0.93**	**0.95**		
5	Lie	0.95	0.96	0.95	95.14%	0.95
	Truth	0.96	0.94	0.95		

TABLE III
PERFORMANCE PROSODIC - CNN

Model	Performance of Prosodic - CNN					
	Classification	**Precision**	**Recall**	**F1-Score**	**Accuracy**	**Area Under Curve (AUC)**
1	Lie	0.55	0.64	0.59	63%	0.63
	Truth	0.71	0.63	0.67		
2	Lie	0.74	0.96	0.84	63%	0.65
	Truth	0.94	0.62	0.75		
3	Lie	0.77	0.86	0.77	65%	0.63
	Truth	0.79	0.59	0.67		
4	Lie	0.55	0.71	0.62	67%	0.68
	Truth	0.79	0.66	0.71		
5	**Lie**	**0.63**	**0.68**	**0.66**	**68%**	**0.69**
	Truth	**0.72**	**0.68**	**0.70**		

MFCC-CNN has a superior performance with a maximum accuracy of 97.13% and AUC value of 0.97, STFT-CNN has an accuracy performance of 95.39% and AUC of 0.95, and Prosodic-CNN has a less optimal performance with an accuracy of 68% and AUC value of 0.69. MFF-CNN has better performance for models in detecting lies with consistent accuracy values starting at epoch 750 to epoch 1200, as well as F1 score, Precision, Recall, and AUC values. The accuracy result is stable between 96% to 97%. Based on Tables I, II, and III, good performance is produced by MFCC-CNN, where the CNN implementation functions as a feature extractor on voice data with several layers. In this research, MFCC is used as a preprocessing stage to produce features that can be recognized by CNN. The use of MFCC features with spectral coefficients in feature extraction can help recognize voice features, namely intonation and pitch, thus improving the performance of the lie recognition model through voice. In addition, to maximize MFCC performance, we used a cepstral coefficient of 20. Using a cepstral coefficient of more than 13 is much better, although many studies use a cepstral coefficient of 13 [36]. The confusion matrix and AUC of the best models from MFCC - CNN are as follows:

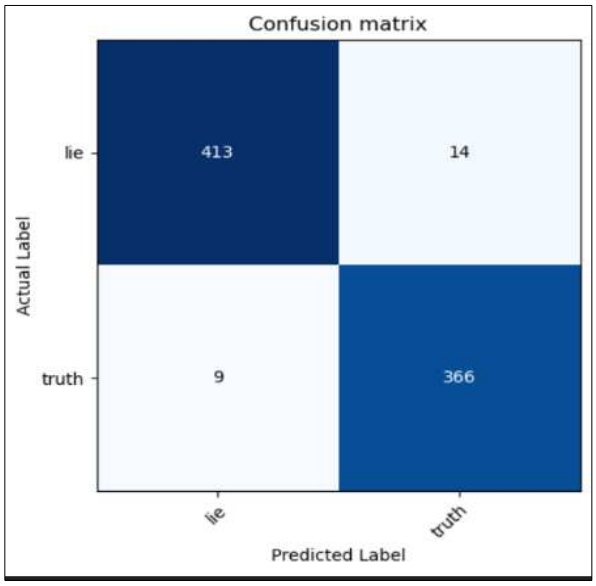

Fig. 7 Confusion matrixes from the MFCC-CNN process

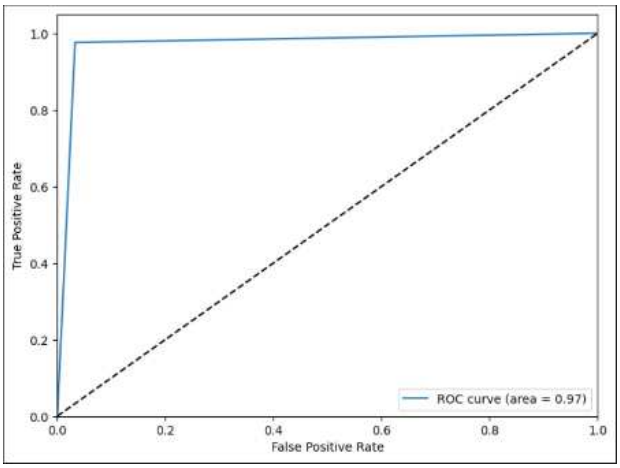

Fig. 8 AUC Value of MFCC-CNN Process

Table IV is a proposed model for detecting lies through sound in the form of signals using CNN.

TABLE IV
PROPOSED MODEL

Layer Type	Output Shape	Parameter
Conv2d(Conv2D)	(None,18,180,112)	1120
Conv2d_1(Conv2D)	(None,14,176,32)	89632
Conv2d_2(Conv2D)	(None,12,174,80)	23120
Max_pooling2d(Maxpooling2D)	(None,6,87,80)	0
Dropout	(None,6,87,80)	0
Flatten	(None,41760)	0
Dense	(None, 224)	9354464
Dropout_1	(None,224)	0
Dense_1	(None,2)	450

This conv2d layer is intended to identify the initial feature of an image, such as lines, edges, or higher contrast. The number of filters is 112, meaning each image after this convolution has 112 features. Conv2d_1 layer, this convolution extracts more features from the image, which is more abstract than the previous layer. The number of filters is 32, and the convolution operation reduces the image's dimensions. Conv2d_2 layer, this convolution continues to improve feature extraction and try to find more complex features. The number of filters is 80. Max_pooling2d This layer is to reduce the dimensions of two-dimensional images in half, height, and width. This helps reduce the complexity and size of the data. Dropouts are used to avoid overfitting and changing product dimensions. Flatten, this layer converts the image into a one-dimensional vector, which is used as input for the next layer. Due to the large number of input (41760) and output neurons (224), this truly connected layer has many parameters. Dropout_1 is used to prevent overfitting. Dense-1 is the last layer of the built model that is fully connected. It produces an output of length 2, which is equivalent to the number of classes present in the binary classification task.

Combining convolution layers, max pooling, dropout, and other connected layers indicates an attempt to build a model that can recognize complex image features and perform accurate classification. In addition, the high number of parameters in the connected layer indicates the model's capacity to learn more complex image data representations.

The different number of parameters present in each layer reflects the complexity and function played by each layer in the model learning process. Due to their role in pattern recognition and more complex features in image data, convolution, and dense layers tend to have more parameters. Dropout and pooling, on the other hand, because they focus on dimensionality reduction, do not add significant parameters.

The results of this study have several important implications. First, this study provides clear evidence that MFCC with classification using CNN can process clear speech features with excellent performance thereby increasing the accuracy and efficiency of lie detection.

Second, this study shows that audio data processed using MFCC-CNN provides better performance in lie detection compared to processed using STFT-CNN and Prosodic-CNN. This could be due to the fact that the process of using the cepstral coefficient of 20 with MFCC provides more detailed

information about the features of the audio data, which allows CNN to produce a better classification in discriminating between honest and lying voices.

VI. Conclusion

In voice data processing, the built CNN model can be used to detect lies in digital voice data and voice data in the form of a spectrogram. The use of audio with a feature extraction process using MFCC as input gives better results in detecting lies. Using the MFCC technique can improve the accuracy of lie detection. The MFCC-CNN model developed can produce good accuracy by using a central coefficient of 20 and adding a dropout layer to avoid overfitting. CNN produces good performance for processing input other than images. The results of the model evaluation show that the model trained on audio sound data has a better performance in detecting lies with an accuracy value of 97.13% and AUC of 0.97, while the model processed using STFT has an accuracy value of 95.39% and AUC of 0.95, the accuracy using Prosodic was 65% with an AUC of 0.69 The average execution time for MFCC-CNN is 20 minutes/epoch while the time execution for STFT-CNN is 22 minutes/epoch and the average execution time for Prosodic-CNN is 21 minutes/epoch. This research has important implications for the development of voice-based security and lie identification systems, maximizing voice processing using MFCC.

For further research, can be optimized by combining CNN architectural models such as ResNet, AlexNet, and other architectures to obtain new models and improve lie detection accuracy. To get a better understanding and generalization of the development of this research, can use a variety of sound datasets in several situations. Besides that, it can explore the combination of features between MFCC and STFT to get new features. Furthermore, research can be developed by testing the performance of the model in real situations, such as investigations, law enforcement, or job interviews. This will help in determining the difficulties and opportunities in implementing lie detection technology in real life.

References

[1] A. R. Bhamare, S. Kathargуppe, and J. Silviya Nancy, “Deep Neural Networks for Lie Detection with Attention on Bio-signals,” 2020 7th International Conference on Soft Computing & Machine Intelligence (ISCMI), Nov. 2020, doi:10.1109/iscmi51676.2020.9311575.

[2] M. Zabcikova, Z. Koudelkova, and R. Jasek, “Concealed Information Detection Using EEG for Lie Recognition by ERP P300 in Response to Visual Stimuli: a Review,” WSEAS Transactions on Information Science and Applications, vol. 19, pp. 171–179, Sep. 2022, doi:10.37394/23209.2022.19.17.

[3] A. Bablani, D. R. Edla, V. Kupilli, and R. Dharavath, “Lie Detection Using Fuzzy Ensemble Approach With Novel Defuzzification Method for Classification of EEG Signals,” IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–13, 2021, doi:10.1109/tim.2021.3082985.

[4] Y. Zhou and F. Bu, “An Overview of Advancements in Lie Detection Technology in Speech,” International Journal of Information Technologies and Systems Approach, vol. 16, no. 2, pp. 1–24, Jan. 2023, doi: 10.4018/ijitsa.316935.

[5] N. Baghel, D. Singh, M. K. Dutta, R. Burget, and V. Myska, “Truth Identification from EEG Signal by using Convolution neural network: Lie Detection,” 2020 43rd International Conference on Telecommunications and Signal Processing (TSP), Jul. 2020, doi:10.1109/tsp49548.2020.9163497.

[6] S. Dodia, D. R. Edla, A. Bablani, and R. Cheruku, “Lie detection using extreme learning machine: A concealed information test based on short-time Fourier transform and binary bat optimization using a novel fitness function,” Computational Intelligence, vol. 36, no. 2, pp. 637–658, Nov. 2019, doi: 10.1111/coin.12256.

[7] A. Curci, T. Lanciano, F. Battista, S. Guaragno, and R. M. Ribatti, “Accuracy, Confidence, and Experiential Criteria for Lie Detection Through a Videotaped Interview,” Frontiers in Psychiatry, vol. 9, Jan. 2019, doi: 10.3389/fpsyt.2018.00748.

[8] V. Gupta, M. Agarwal, M. Arora, T. Chakraborty, R. Singh, and M. Vatsa, “Bag-of-Lies: A Multimodal Dataset for Deception Detection,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2019, doi: 10.1109/cvprw.2019.00016.

[9] D. Pasquali, J. Gonzalez-Billandon, A. M. Aroyo, G. Sandini, A. Sciutti, and F. Rea, “Detecting Lies is a Child (Robot)’s Play: Gaze-Based Lie Detection in HRI,” International Journal of Social Robotics, vol. 15, no. 4, pp. 583–598, Nov. 2021, doi: 10.1007/s12369-021-00822-5.

[10] H. U. D. Ahmed, U. I. Bajwa, F. Zhang, and M. W. Anwar, “Deception Detection in Videos using the Facial Action Coding System,” pp. 0–2, 2021.

[11] V. Pérez-Rosas, M. Abouelenien, R. Mihalcea, and M. Burzo, “Deception Detection using Real-life Trial Data,” Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, Nov. 2015, doi: 10.1145/2818346.2820758.

[12] Z. Wu, B. Singh, L. Davis, and V. Subrahmanian, “Deception Detection in Videos,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, Apr. 2018, doi: 10.1609/aaai.v32i1.11502.

[13] H. Nasri, W. Ouarda, and A. M. Alimi, “ReLiDSS: Novel lie detection system from speech signal,” *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA*, vol. 0, 2016, doi: 10.1109/AICCSA.2016.7945789.

[14] M. Delgado-Herrera, A. Reyes-Aguilar, and M. Giordano, “What Deception Tasks Used in the Lab Really Do: Systematic Review and Meta-analysis of Ecological Validity of fMRI Deception Tasks,” Neuroscience, vol. 468, pp. 88–109, Aug. 2021, doi: 10.1016/j.neuroscience.2021.06.005.

[15] T. K. Ying-Li Tian and and Jeffrey F.Cohn, “Chapter 11. Facial Expression Analysis,” *J. Infect. Dis.*, vol. 174, no. 4, pp. 835–838, 2013.

[16] M. Arsal, B. Agus Wardijono, and D. Anggraini, “Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN,” Jurnal Nasional Teknologi dan Sistem Informasi, vol. 6, no. 1, pp. 55–63, Jun. 2020, doi: 10.25077/teknosi.v6i1.2020.55-63.

[17] M. Owayjan, A. Kashour, N. Al Haddad, M. Fadel, and G. Al Souki, “The design and development of a Lie Detection System using facial micro-expressions,” 2012 2nd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), Dec. 2012, doi: 10.1109/ictea.2012.6462897.

[18] D. Barsever, S. Singh, and E. Neftci, “Building a Better Lie Detector with BERT: The Difference Between Truth and Lies,” 2020 International Joint Conference on Neural Networks (IJCNN), Jul. 2020, doi: 10.1109/ijcnn48605.2020.9206937.

[19] I. Lakshan, L. Wickramasinghe, S. Disala, S. Chandrasegar, and P. S. Haddela, “Real Time Deception Detection for Criminal Investigation,” 2019 National Information Technology Conference (NITC), Oct. 2019, doi: 10.1109/nitc48475.2019.9114422.

[20] J. Immanuel, A. Joshua, and S. T. George, “A Study on Using Blink Parameters from EEG Data for Lie Detection,” 2018 International Conference on Computer Communication and Informatics (ICCCI), Jan. 2018, doi: 10.1109/iccci.2018.8441238.

[21] S. Kamran Haider, M. I. Daud, A. Jiang, and Z. Khan, “Evaluation of P300 based Lie Detection Algorithm,” *Electr. Electron. Eng.*, vol. 2017, no. 3, pp. 69–76, 2017, doi: 10.5923/j.eee.20170703.01.

[22] J. Gao, H. Tian, Y. Yang, X. Yu, C. Li, and N. Rao, “A Novel Algorithm to Enhance P300 in Single Trials: Application to Lie Detection Using F-Score and SVM,” PLoS ONE, vol. 9, no. 11, p. e109700, Nov. 2014, doi: 10.1371/journal.pone.0109700.

[23] B. Singh, P. Rajiv, and M. Chandra, “Lie detection using image processing,” 2015 International Conference on Advanced Computing and Communication Systems, Jan. 2015, doi: 10.1109/icaccs.2015.7324092.

[24] W. Khan, K. Crockett, J. O’Shea, A. Hussain, and B. M. Khan, “Deception in the eyes of deceiver: A computer vision and machine learning based automated deception detection,” Expert Systems with Applications, vol. 169, p. 114341, May 2021, doi:10.1016/j.eswa.2020.114341.

[25] E. P. Fathima Bareeda, B. S. Shajee Mohan, and K. V. Ahammed Muneer, “Lie Detection using Speech Processing Techniques,”

Journal of Physics: Conference Series, vol. 1921, no. 1, p. 012028, May 2021, doi: 10.1088/1742-6596/1921/1/012028.

[26] A. Kusnadi, I. M. O. Widyantara, and L. Linawati, "Deteksi Kebohongan Berdasarkan Fitur Fonetik Akustik," Majalah Ilmiah Teknologi Elektro, vol. 20, no. 1, p. 113, Mar. 2021, doi:10.24843/mite.2021.v20i01.p13.

[27] Y. Yohannes and R. Wijaya, "Klasifikasi Makna Tangisan Bayi Menggunakan CNN Berdasarkan Kombinasi Fitur MFCC dan DWT," JATISI (Jurnal Teknik Informatika dan Sistem Informasi), vol. 8, no. 2, pp. 599–610, Jun. 2021, doi: 10.35957/jatisi.v8i2.470.

[28] J. Li, X. Zhang, L. Huang, F. Li, S. Duan, and Y. Sun, "Speech Emotion Recognition Using a Dual-Channel Complementary Spectrogram and the CNN-SSAE Neutral Network," Applied Sciences, vol. 12, no. 19, p. 9518, Sep. 2022, doi:10.3390/app12199518.

[29] N. D. Miranda, L. Novamizanti, and S. Rizal, "Convolutional Neural Network pada Klasifikasi Sidik Jari Menggunakan Resnet-50," Jurnal Teknik Informatika (Jutif), vol. 1, no. 2, pp. 61–68, Dec. 2020, doi:10.20884/1.jutif.2020.1.2.18.

[30] F. Cai, L. Ma, Y. Lu, Y. Hu, and S. Su, "Combining Artificial Intelligence with Traditional Chinese Medicine for Intelligent Health Management," ... *Mach. Learn.*, 2021.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[32] A. B. Gumelar et al., "Human Voice Emotion Identification Using Prosodic and Spectral Feature Extraction Based on Deep Neural Networks," 2019 IEEE 7th International Conference on Serious Games and Applications for Health (SeGAH), Aug. 2019, doi:10.1109/segah.2019.8882461.