Distress Detection Using A Hybrid SVM - CNN Classifier

Modha Varsha
Dept. of Computer Science and Engineering
Global Academy of Technology
Bengaluru, India
modhavarsha1310@gmail.com

Yukthi R Aithal
Dept. of Computer Science and Engineering
Global Academy of Technology
Bengaluru, India
yukthiraithal@gmail.com

Sufia Fathima
Dept. of Computer Science and Engineering
Global Academy of Technology
Bengaluru, India
sufiafathima17@gmail.com

Snigdha Sen
Dept. of Computer Science and Engineering
Global Academy of Technology
Bengaluru, India
snigdha.sen@gat.ac.in

***Abstract*— Due to the escalating frequency of reported crimes, there has been a surge in research endeavors focused on various approaches to enhance monitoring and surveillance techniques. As opposed to vision-based applications, which are currently the most used framework for monitoring purposes, audio-based systems can be more flexible and relatively less intrusive. While current research studies predominantly utilize images as the primary input for Deep Learning (DL) algorithms, it is worth noting that sound can also serve as a valuable source of input for these models. In this paper, we propose and develop a novel hybrid deep learning model for identifying and detecting people in distress by their screams. The working of our proposed system is built by integrating sound detection module and DL model which will help us to detect if a person is in distress or not. The system uses hybridization concept consisting of Support Vector Machine (SVM) and Convolutional Neural Network (CNN) models, with the audio snippet undergoing 3 levels of classification, with the accuracy of each level found to be 93%, 100% and 92% respectively. The audio data is finally classified as an instance of distress or no-distress.**

Keywords*— *Sound Detection, Audio Classification, Support Vector Machine, Convolutional Neural Networks, Distress Detection.

I. INTRODUCTION

Crime poses a significant social challenge that we encounter in our daily existence. We still don’t have systems which are fully functional to alert someone and get help when a person is in distress. There is a need to develop a system that gets activated on its own when a person is in distress and gets help. This system can be developed using Sound Event Detection and Machine Learning which act as the main domains.

Here, we build a model using a hybrid architecture of Support Vector Machine and Convolution Neural Network that can take the input of an audio snippet, initially classify the input as an instance of a vocal or a non-vocal event then, further classify the vocal event as an instance of normal conversation or shouting using SVM. Advancing in architecture, the CNN framework should classify an instance of shouting as distressed or not-distressed. If any instance is eventually classified as distressed, the system should send an alert to initiate help.

The system includes the audio processing and feature extraction module and classification module. The system will take audio data as the input, perform pre-processing using spectral analysis algorithms to obtain their respective compact representations. The dataset is subsequently divided into a training set and a testing set. The two models are trained and tested with the appropriate datasets. Once a threshold accuracy is obtained from both model's classification process, they are combined to obtain a full-fledged framework capable of detecting emotional distress in vocal audio.

II. LITERATURE SURVEY

A. Emotion Detection

In recent years, numerous studies have been undertaken to identify non-speech sounds, owing to their practical applications in various domains. Multiple researchers have applied machine learning algorithms to classify audio. Waldekar and Saha [1] created a fused system architecture. This method employed frame-level statistics of the recognized spectral features as input to the Support Vector Machine (SVM), showcasing the system's outstanding performance. Souli and Lachiri [2] used SVM and scattering characteristics to classify audio. Discriminating between events is made possible by the ability to represent nonstationary signals, which includes filters for time and rhythms. Furthermore, the method's classification accuracy performed well in quiet surroundings as opposed to noisy ones. In the paper by Khamparia et al. [3] environmental sounds were classified using a two-layer deep Convolutional Neural Network (CNN) and a Tensor Deep Stacking Network (TDSN). They created a spectrogram using audio datasets ESC-10 [4] and ESC-50 [5] using MATLAB's built-in "spectrogram ()" function. Bui et al. [6] used Deep CNN to analyse traffic flow in the metropolitan road network. They produced a time-frequency representation using log-Mel spectrograms and Mel-frequency Cepstral coefficients (MFCC) (TFR). They then used CNN, consisting of five layers and four max-pooling procedures regarding emotion detection. The paper by W. Zheng et al. [7] consisted of a Convolutional Neural Network (CNN) architecture that was developed to carry out emotion identification on labelled data. The final experimental outcomes demonstrated that their proposed approach outperformed SVM classification. H. Meng et al. [8] utilized the feature vector of the raw speech signal, which is composed of the log Mel spectrum, delta, and deltas-deltas, as the input for their suggested model. They created ADRNN (dilated CNN with residual block and BiLSTM) networks with a unique architecture to identify speech emotion, and the results indicate that these networks perform better overall than ACRNN networks. ADRNN networks also outperform other well-known feature representations and techniques in terms of average accuracy.

B. *Danger Detection*

In their study on women's security gadgets, Khandelwal et al. [9] developed a system where body temperature and pulse readings are recorded, and an alarm is set off when they exceed a certain threshold. Using machine learning to analyse the pattern of typical data, the threshold is adjusted for each individual separately. In their "Surakkha" apparatus, Bhardwaj et al. [10] to detect threats, a system was designed to analyse voice patterns, switches, and shocks.. They created a special tool for this. When users are in danger, certain words will be programmed into their gadgets. It is a voice recognition system that is hardware-based. A few methods to spot danger for women were demonstrated by Hossain et al. [11]. For threat detection, SOS buttons, voice commands, shaking of phones, and concealed camera recordings are used. When a woman is in danger, she must either cry "Help Me!" or push the SOS button on her phone. The application detects a threat because of these activities. These commands are recognised using Google Voice Recognition software. But when someone is in danger, they might not be able to speak clearly due to their state of stress. So, it is preferable to recognise danger through cries or shouts rather than from words.

III. Dataset Description

The datasets used are from various sources to diversify the training and test data. The sources include People's Speech Dataset [12], it is a large and diverse English speech recognition corpus that contains over 30,000 hours of transcribed speech. It is one of the largest speech datasets available today and is licensed under the Creative Commons Attribution-ShareAlike (CC-BY-SA) and Creative Commons Attribution (CC-BY) licenses, which allows for academic and commercial usage of the dataset. With its size and permissive license, the People's Speech Dataset is a valuable resource for training speech-to-text systems and advancing research in the field of speech recognition.

Freesound [13] is a collaborative database and repository of Creative Commons licensed audio samples, which allows users to upload and share their own sounds with others. It is a non-profit organization and has over 8 million registered users. The website currently hosts more than 500,000 sounds and effects, covering a diverse range of topics and genres such as field recordings, soundscapes, music, voice recordings, and synthesized sounds. With its wide variety of sound samples and permissive licensing, Freesound is a valuable resource for sound designers, musicians, filmmakers, and anyone else in need of high-quality sound effects and audio samples.

The UrbanSound8K [14] dataset is a labelled dataset of 8732 sound excerpts which are less than or equal to a time length of 4s. It consists of urban sounds, categorized into 10 classes such as air_conditioner, car_horn, children_playing, dog_bark, drilling, engine_idling, gun_shot, jackhammer, siren, and street_music. The dataset was generated by utilizing field recordings obtained from www.freesound.org. The files were organized into ten folds, simplifying the comparison and reproduction of automatic classification outcomes. All audio files are in WAV format, maintaining the same sampling rate, bit depth, and number of channels as the original files uploaded to Freesound, which may vary across different files. The UrbanSound8K dataset is commonly used in machine learning and audio classification research to develop and evaluate sound classification algorithms.

We have also used excerpts from various YouTube videos whose audio files were extracted and cut short to create more instances of data to help build the models.

IV. Methodology

A. *Framework Used*

The system follows a hybrid architecture consisting of three phases that are built using SVM (Fig. 1) and CNN (Fig. 2) models. Initially, when the dataset is fed into the system, the first phase uses the SVM, which is a two-phase model that detects non-speech audio input [15]. SVMs are effective at classifying sounds in our model and have the benefit of adaptability, as numerous kernels are permitted for classification. In the first phase, the data is classified as vocal or non-vocal events using the SVM that works on the linear kernel. In the second phase, screams or shouts are detected with the MFCC, which is entered in the RBF-kernel based SVM. Once the second phase identifies the data as a scream or shout, the sound is further processed and turned into graphical images for further classification. Mel spectrograms are the pictorial representations of this data and are then used to accurately classify data as distressed or not using the CNN model, as represented in Fig. 3.

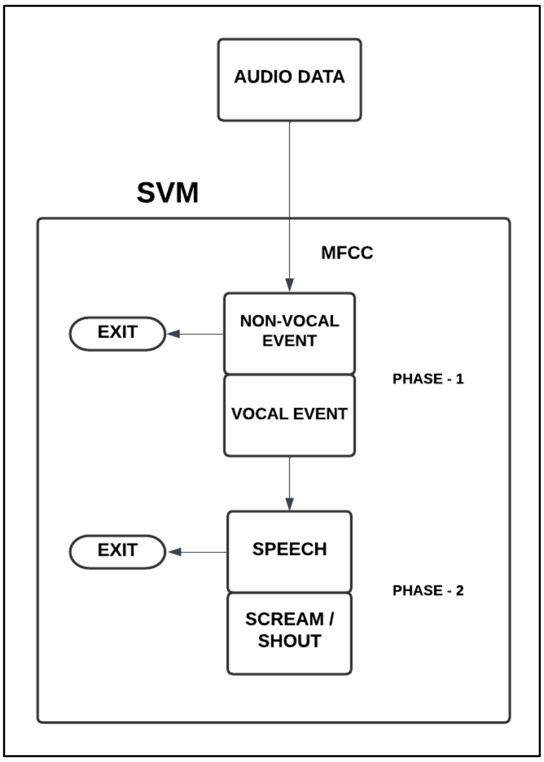


Fig. 1. SVM Architecture

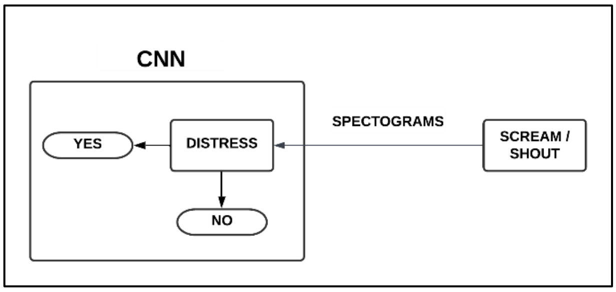


Fig. 2. CNN Architecture

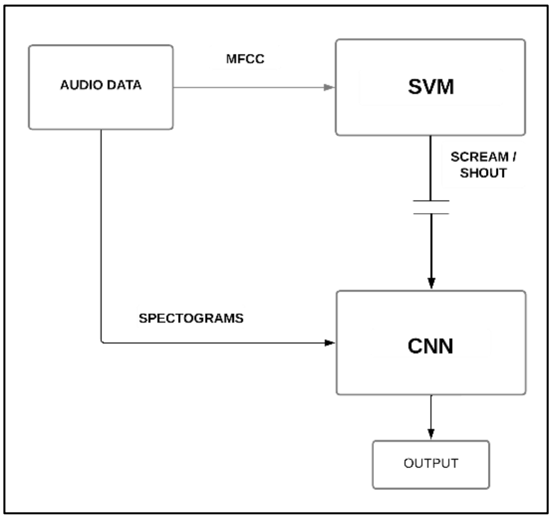


Fig. 3. System Architecture

B. *Audio Processing*

At the initial stage, upon receiving the audio input signal, it is necessary to pre-process the signal into a format that can be comprehended by the deep learning algorithms. This process consists of a few steps that are common to processing the input for the whole framework.

First, the input signal is framed so that the number of samples generated for each second is equal. [16] Framing is a technique where multiple channels are combined into a series of audio frames, where each frame contains one sample for each channel. A sample refers to a numerical value that represents the amplitude of the sound waveform at a specific moment in time. The sample rate, also known as the sampling frequency, determines the number of samples recorded or generated per second.

However, simply cutting the signal into segments can introduce undesirable artifacts at the segment boundaries due to abrupt changes in the signal. This is known as the "leakage" problem. To overcome this issue, windowing is used. [17] Windowing is a commonly used technique in signal processing, particularly in the analysis of non-stationary signals. The goal of windowing is to break up a long signal into shorter, overlapping segments that can be analysed more easily. A window function is typically applied to each segment. The window function is a smooth function that gradually tapers the edges of the segment to zero, thereby reducing the abruptness of the segment boundary. Commonly used window functions include the Hamming, Hanning, and Blackman windows. Multiplying the signal by the window function effectively weights the signal within the window, giving greater emphasis to the central portion of the segment and reducing the contribution of the edges. The resulting windowed signal is smoother and more well-behaved, making it easier to analyse.

Next, Fast Fourier Transform (FFT) [18] is applied to the windowed signal. FFT is used to transform the signals into the spectral domain. The spectrum obtained is transformed into the Mel-Scale using the Mel Frequency Transform. The Mel-Scale is a perceptual frequency scale that aims to replicate the human auditory system's sensitivity to various frequency regions. It is based on the observation that the perception of pitch is not linear with respect to frequency, and the Mel-Scale provides a nonlinear mapping of frequency to a scale that is more consistent with human perception. The obtained Mel Spectrum is the input for the CNN architecture.

The Mel Spectrum is further converted into Mel Frequency Cepstrum (MFC). The MFC [19] is computed by first computing the power spectrum of a short-time windowed segment of the audio signal using techniques such as the Fourier transform. Then, the power spectrum is converted to a log scale to account for the logarithmic nature of human perception. Next, the Mel Filterbank is employed, which comprises a collection of triangular filters that are evenly spaced on the Mel-Scale and overlapped in frequency. The log power spectrum is then multiplied with each Mel Filterbank to compute the Mel Energy for each filter. Finally, a discrete cosine transform (DCT) is applied to the Mel Energies to obtain the MFC coefficients, which are typically used as features for various audio and speech processing tasks, such as speaker recognition, speech recognition, and audio classification. MFCCs (Mel-Frequency Cepstral Coefficients) are the coefficients that make up the MFC, typically capturing the most important spectral features of the audio signal. This MFCC is the input to the SVM architecture. Fig. 4 shows the audio processing steps.

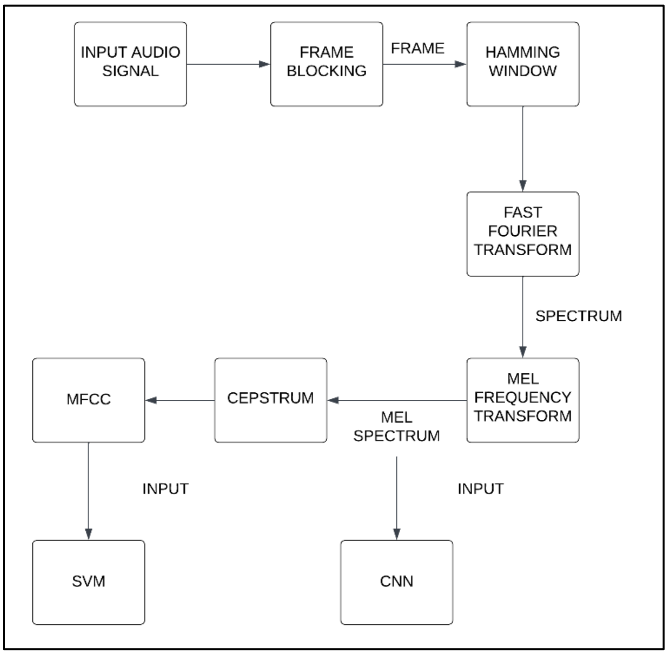


Fig. 4. Audio processing done parallelly for the hybrid architecture.

Fig. 5 and Fig. 6 are snapshots of the Mel Spectograms for different vocal events.

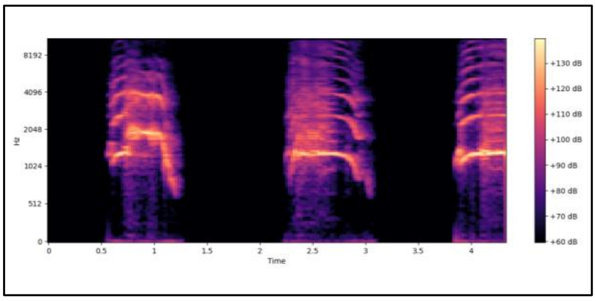


Fig. 5. Spectogram of an Audio Snippet classified as Distressed.

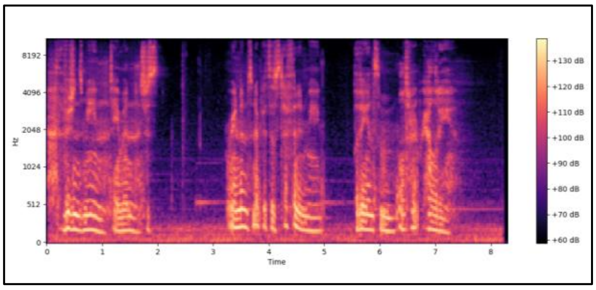


Fig. 6. Spectogram of an Audio Snippet classified as Not- Distressed.

V. Results

The goal of this work is to apply machine learning and detect an instance of distress. Python library matplotlib has been used to generate graphs. Fig. 7 and Fig. 8 represent the Confusion Matrix and Precision, Recall, F1 Score of the SVM Phase 1 respectively. Here, out of 58 audio snippets from the test data, the number of occurrences of Vocal Events was 32. The model was successful in predicting 91% of the Vocal Events (represented by class label 2) out of the total test cases accurately. The Non-Vocal events (represented by class label 1) were predicted with an accuracy of 96%. The overall exactness of the model is found to be 93% on the test dataset. Fig. 9 and Fig. 10 represent the Confusion Matrix and Precision, Recall, F1 Score of the SVM Phase 2 respectively. There were a total of 34 audio snippets in the test data, of which the number of Scream Events(represented by class label 2)was 20. The Scream Events were predicted with an accuracy of 100% which was the same as the accuracy for the Speech Events (represented by class label 1). This made the SVM Phase 2 model have an accuracy of 100%. Confusion Matrix and Precision, Recall, F1 Score of the CNN are represented by Fig. 11 and Fig. 12 respectively. The Distressed Events (represented by class label 0) constituted 13 out of 37 audio snippets. They were classified with an accuracy of 85% while the Not-Distressed Events (represented by class label 1) were predicted with an accuracy of 96%. The CNN model had an overall exactness of 92%.

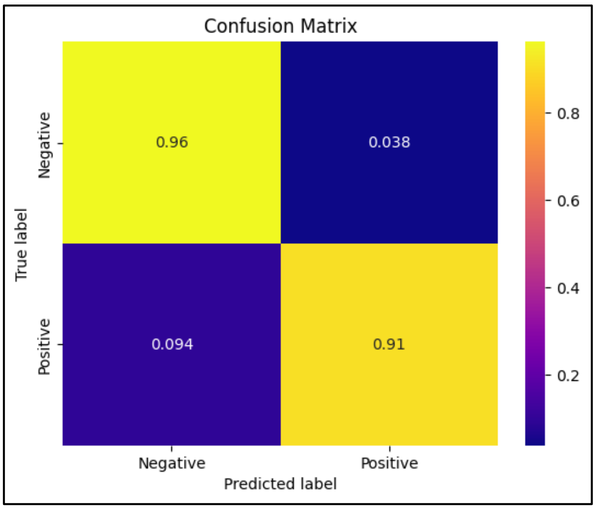
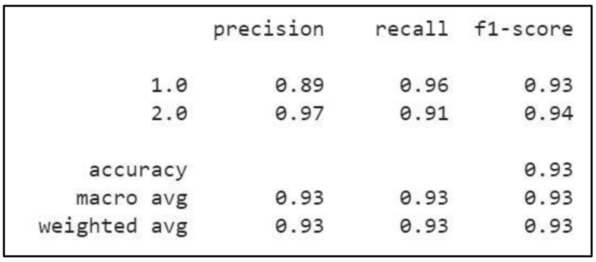


Fig. 7. Confusion Matrix of SVM Phase 1



```
              precision    recall  f1-score

         1.0       0.89      0.96      0.93
         2.0       0.97      0.91      0.94

    accuracy                           0.93
   macro avg       0.93      0.93      0.93
weighted avg       0.93      0.93      0.93
```

Fig. 8. Precision, Recall and F1 Score of SVM Phase 1

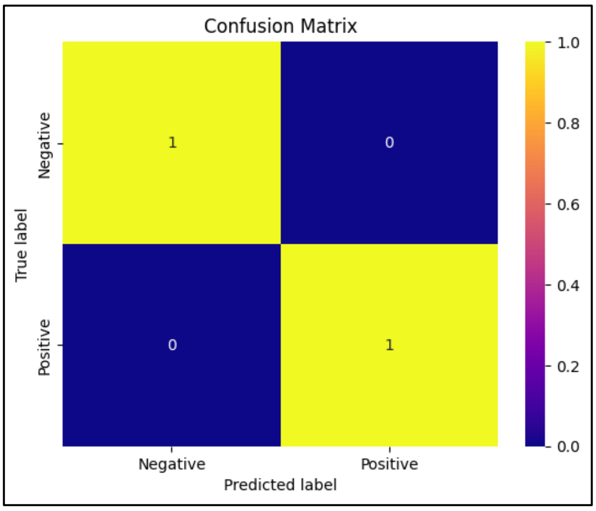
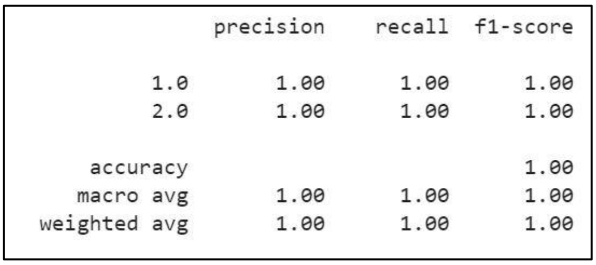


Fig. 9. Confusion Matrix of SVM Phase 2



```
              precision    recall  f1-score

         1.0       1.00      1.00      1.00
         2.0       1.00      1.00      1.00

    accuracy                           1.00
   macro avg       1.00      1.00      1.00
weighted avg       1.00      1.00      1.00
```

Fig. 10. Precision, Recall and F1 Score of SVM Phase 2

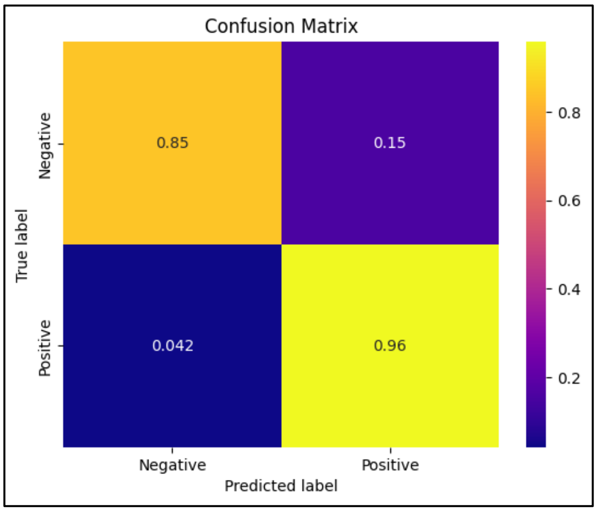
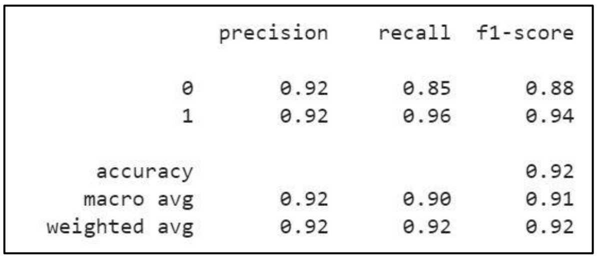


Fig. 11. Confusion Matrix of CNN



```
              precision    recall  f1-score

           0       0.92      0.85      0.88
           1       0.92      0.96      0.94

    accuracy                           0.92
   macro avg       0.92      0.90      0.91
weighted avg       0.92      0.92      0.92
```

Fig. 12. Precision, Recall and F1 Score of CNN

Fig. 13 is a graph of Accuracy vs Loss; the graph indicates decrease in loss with increase in accuracy. This means the model shows the possibility of being a good fit with more training involving more diverse data points. The plotting of loss in test data vs train data is shown in Fig. 14. We can see that the loss is similar in both cases. The plot of variation in accuracies of Test Data vs Train Data is represented in Fig. 15. The accuracy of the Train Data fluctuates initially before increasing and stabilizing while the accuracy of the Test Data is stable before increasing. Both the accuracies are almost the same at the end, this implies the model has understood the dataset well.

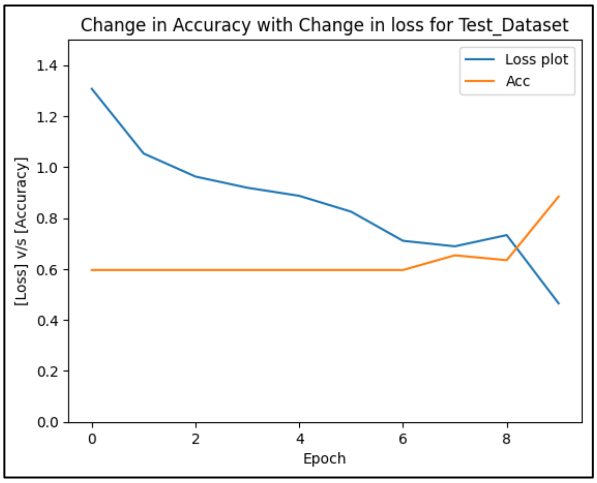


Fig. 13. Graph of Accuracy vs Loss for Test Data

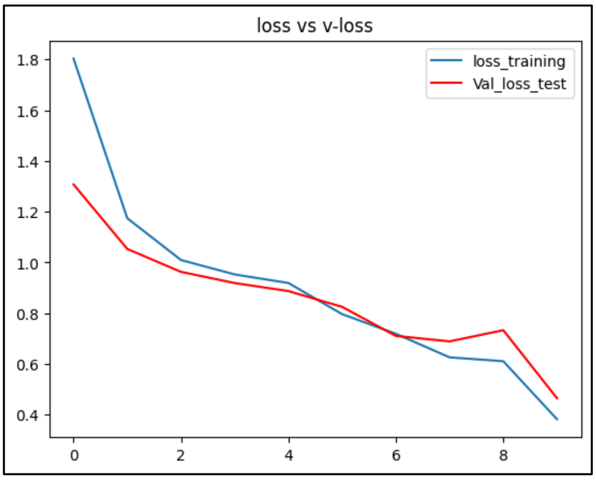


Fig. 14. Graph of Loss in Test Data vs Loss in Training Data

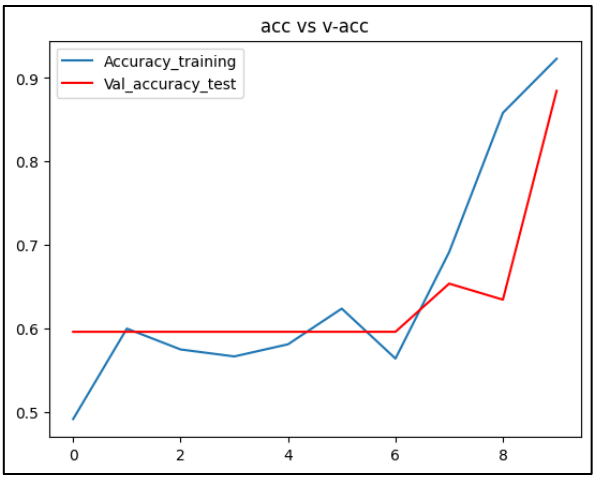


Fig. 15. Graph of Test Data Accuracy vs Training Data Accuracy

VI. Conclusion

Sound processing is used to identify Distressed vocal events. We divided the system into three phases. MFCC arrays and Mel Spectograms are used as input vectors for the SVM phases and CNN respectively. Audio processing initially consists of framing and windowing to break the audio snippets into equal sizes. Next, the snippets are converted into Mel Spectograms by using Fast Fourier Transform and Mel Frequency Transform. These Spectograms are then converted to MFCC arrays. The hybrid model is then trained and tested at each phase. The final accuracy was 92%. This implies the model is efficient enough to detect probable distress events. The system's effectiveness can be further enhanced by integrating it with a front-end application that can be deployed in real-time scenarios. This integration can contribute to improving safety measures and reducing crime incidents.

References

[1] Waldekar, S, Saha, G.: Classification of audio scenes with novel features in a fused system framework. Digit. Signal Process. 75, 71–82 (2018).

[2] Souli, S., Lachiri, Z.: Audio sounds classification using scattering features and support vector machines for medical surveillance. Appl. Acoust. 130, 270–282 (2018).

[3] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey, and P. Tiwari, “Sound classification using convolutional neural network and

tensor deep stacking network," IEEE Access, vol. 7, pp. 7717–7727, 2019.

[4] Nath, Surabhi. "Hear Her Fear: Data Sonification for Sensitizing Society on Crime Against Women in India." IndiaHCI'20: Proceedings of the 11th Indian Conference on Human-Computer Interaction. 2020.

[5] Shankhdhar, Ashutosh, Vinay Kumar, and Yash Mathur. "Human Scream Detection Through Three-Stage Supervised Learning and Deep Learning." Inventive Systems and Control: Proceedings of ICISC 2021. Springer Singapore, 2021.

[6] K.-H. N. Bui, H. Oh, and H. Yi, "Traffic density classification using sound datasets: An empirical study on traffic flow at asymmetric roads," IEEE Access, vol. 8, pp. 125671–125679, 2020.

[7] W. Q. Zheng, J. S. Yu, and Y. X. Zou, ''An experimental study of speech emotion recognition based on deep convolutional neural networks,'' in Proc. IEEE Int. Conf. Affect. Comput. Intell. Interact., Sep. 2015, pp. 827–831.

[8] Meng, Hao, et al. "Speech emotion recognition from 3D log-mel spectrograms with deep learning network." IEEE access 7 (2019): 125868-125881.

[9] T. Khandelwal, M. Khandelwal, P. S. Pandey, et al., "Women safety device designed using iot and machine learning," in 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), pp. 1204–1210, IEEE, 2018.

[10] N. Bhardwaj and N. Aggarwal, "Design and development of "suraksha"- a women safety device," International Journal of Information & Computational Technology, vol. 4, no. 8, pp. 787–792, 2014.

[11] M. E. Hossain, M. W. Rahman, M. T. Islam, and M. S. Hossain, "Manifesting a mobile application on safety which ascertains women salus in bangladesh," International Journal of Electrical and Computer Engineering, vol. 9, no. 5, p. 4355, 2019.

[12] https://mlcommons.org/en/peoples-speech/.

[13] https://en.wikipedia.org/wiki/Freesound.

[14] https://www.kaggle.com/datasets/chrisfilo/urbansound8k.

[15] Shankhdhar, A., Rachit, Kumar, V., Mathur, Y. (2021). Human Scream Detection Through Three-Stage Supervised Learning and Deep Learning. In: Suma, V., Chen, J.IZ., Baig, Z., Wang, H. (eds) Inventive Systems and Control. Lecture Notes in Networks and Systems, vol 204. Springer, Singapore. https://doi.org/10.1007/978-981-16-1395-1_28.

[16] developer.mozilla.org/en-US/docs/Web/Media/Formats/Audio_concepts.

[17] https://wiki.aalto.fi/display/ITSP/Windowing.

[18] https://www.sciencedirect.com/topics/engineering/fast-fourier-transform-algorithm#:~:text=The%20FFT%20algorithm%20is%20one,are%20performed%20using%20an%20FFT.

[19] en.wikipedia.org/wiki/Mel-frequency_cepstrum.

[20] Sen, S., Agarwal, S., Chakraborty, P., & Singh, K. P. (2022). Astronomical big data processing using machine learning: A comprehensive review. Experimental Astronomy, 53(1), 1-43.

[21] Sen, S., Thejas, B. K., Pranitha, B. L., & Amrita, I. (2021). Analysis, visualization and prediction of COVID-19 pandemic spread using machine learning. In Innovations in Computer Science and Engineering: Proceedings of 8th ICICSE (pp. 597-603). Springer Singapore.

[22] Bhargav, K. M., Bhat, A., Sen, S., Reddy, A. V. K., & Ashrith, S. D. (2023). Voice-Based Intelligent Virtual Assistant for Windows. In Innovations in Computer Science and Engineering: Proceedings of the Tenth ICICSE, 2022 (pp. 491-500). Singapore: Springer Nature Singapore.

[23] Sahana, D., Varsha, K. S., Sen, S., & Priyanka, R. (2023). A CNN-Based Approach for Facial Emotion Detection. In Soft Computing: Theories and Applications: Proceedings of SoCTA 2022 (pp. 1-10). Singapore: Springer Nature Singapore.