Wavelet Multiresolution Analysis Based Speech Emotion Recognition System Using 1D CNN LSTM Networks

Aditya Dutt, *Graduate Student Member, IEEE*, and Paul Gader, *Fellow, IEEE*

***Abstract*—Speech Emotion Recognition (SER) is the task of recognizing a speaker's emotional state from speech. SER plays a significant role in Human-Computer Interaction and psychological assessment. Several kinds of time-frequency representations like spectrograms, mel-frequency cepstrum coefficients (MFCCs), and mel-spectrograms are commonly used to develop an SER system. These representations use the Fast Fourier Transform (FFT) to convert the time domain signal to the frequency domain. However, the FFT has one fundamental limitation due to the uncertainty principle, which does not simultaneously allow a good resolution in both time and frequency domains. On the other hand, the multiresolution property of wavelets can provide a good localization in both time and frequency domains. Therefore, this article investigates the competency of the wavelet transforms for SER. We propose a Wavelet based Deep Emotion Recognition (WaDER) method using an autoencoder and 1D convolutional neural network (CNN) and long short-term memory (LSTM) networks. The autoencoder is used to perform the dimensionality reduction of the wavelet features then the latent space is used to classify the emotions using the 1D CNN-LSTM model. We conducted a Monte-Carlo K-fold validation using the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset. For speaker-dependent (SD) experiments, we achieved an unweighted accuracy (UA) of 81.45% and a weighted accuracy (WA) of 81.22%. The results of the experiments on the RAVDESS dataset show that the proposed method performs better than the state-of-the-art methods, which use other time-frequency representations.**

***Index Terms*—Speech emotion recognition, wavelets, long short-term memory (LSTM), convolutional neural networks (CNN), autoencoders, dimensionality reduction.**

I. Introduction

EMOTION recognition plays a vital role in Human-Computer Interaction. We can use speech emotion recognition (SER) to make the conversation between machines and humans more intelligent. SER also has applications in healthcare. It can be used to identify psychological disorders which can mitigate the risk of suicidal behaviors. Virtual Emotion AI chatbots can provide personalized therapy by interacting with patients online. SER can be used for emotional speech generation as well. A London-based company, DeepZen, partnered with NVIDIA to develop a deep learning model that can generate human-like emotional speech for audiobooks.

Speech emotions have a tremendous amount of acoustic variance. Therefore, the first step is to identify distinguishable and salient features in a voice segment to get a better recognition rate in SER. Traditionally, researchers use features like mel-frequency cepstrum coefficients (MFCC) [1], spectrograms, mel-spectrograms, energy, fundamental frequency (F0), spectral centroid, and zero-crossing rate. Additionally, many features can be handcrafted using the time-domain features' statistics. Over the past several decades, Hidden Markov Models (HMM) [2], [3], Gaussian Mixture Models (GMM), and Support Vector Machines (SVM) [4], [5], [6] have been used for SER. Various researchers have leveraged the combination of different features to get a better recognition rate. Yogesh et al. [7] extracted biospectral and biocoherence features from glottal and speech waveforms. Seehapoch et al. [4] used features like fundamental frequency, energy, zero-crossing rate, and linear prediction coefficients (LPC) to train an SVM model. Although these models require fewer parameters and are highly interpretable, they have a few limitations when capturing complex non-linear patterns from data.

Deep learning has made a significant improvement in this field. Researchers have developed variations of CNNs and LSTMs to model the spatial and temporal dependencies from the input features. Deep learning can extract salient and discriminative information from the input features to perform an accurate classification. Spectrograms, Chromagrams, and MFCCs can be fed to a CNN or LSTM network as inputs. In 2018, Zhang et al. [8] proposed to use 3-channel log-mel spectrograms as features to train their Deep Convolutional Neural Network. The 3-channels of the mel-spectrogram were static, delta, and delta-delta. The delta and delta-delta are the first and second derivatives of the signal. This representation resembles an RGB (red, green, and blue channels) image. In 2019, Zhao et al. [9] used a local feature learning block (LFLB) and an LSTM model to learn features from raw audio and log-mel spectrograms.

Manuscript received 5 July 2022; revised 30 November 2022; accepted 6 May 2023. Date of publication 17 May 2023; date of current version 26 May 2023. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Abdelrahman Mohamed. *(Corresponding author: Aditya Dutt.)*

Aditya Dutt is with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: aditya.dutt@ufl.edu).

Paul Gader is with the Department of Computer and Information Science and Engineering, and the Department of Environmental Engineering Sciences, University of Florida, Gainesville, FL 32611 USA (e-mail: pgader@ufl.edu).

Digital Object Identifier 10.1109/TASLP.2023.3277291

Similarly, Khorram et al. [10] proposed a dilated CNN-based model [11] to capture the long-term dependencies from data while keeping the number of parameters low.

Recently, the attention mechanism gained much attention as it can focus on the relevant parts of the input to make a decision. Initially, the attention mechanism was introduced by Bahdanau et al. [12] for the machine translation task. However, it has been widely applied for classification purposes as well. Xie et al. [13] proposed an attention-based LSTM model for SER to utilize the difference in emotional saturation between multiple time frames. Mirsamadi et al. [14] also used a bidirectional LSTM with an attention mechanism to focus on the emotionally salient parts of speech. Xu et al. [15] proposed a method called Head Fusion based on a multi-head attention mechanism for speech emotion recognition. They used MFCC features after dividing each sample into multiple fixed-size utterances. They also experimented with different types of noise injections which increased the robustness of the model. Yu et al. [16] used IS09 and mel-spectrograms as features and trained them using an attention-based LSTM model. Some methods used raw audio to perform SER [17]. Since the human auditory system is designed to perceive the frequency and amplitude of sound [18], the focus of this article is to utilize frequency-based features instead of raw audio.

However, most of these methods utilize spectrograms, MFCCs, and mel-spectrograms which use the FFT to convert the time domain signal to the frequency domain. But due to the uncertainty principle, FFT cannot simultaneously get a good resolution in both time and frequency domains. FFTs use a fixed-size window to capture different frequencies. The higher frequencies require a smaller window and the lower frequencies require a bigger window. However, the multiresolution property of the wavelet transforms provides localization in both time and frequency domains simultaneously. In the early 80 s, orthogonal wavelets were discovered by Strömberg [19]. In 1982, Alex Grossman and Jean Morlet developed a continuous wavelet transform [20], [21] for seismic frequency analysis. With the advent of deep learning, wavelet transforms once again gained attention for time series classification. Some researchers have also investigated wavelets for SER tasks using deep learning approaches. Wavelet transform features can have very high dimensionality. Earlier it was challenging to train a neural network using such data due to the computational limits. However, high-dimensional data can be used more easily nowadays to train a deep neural network due to the increased computational power.

Zhiyan et al. [22] used wavelet features and an HMM model to classify Chinese emotional speech. Shegokar et al. [23] used the continuous wavelet transform (CWT) and prosodic coefficients as features and classified them using an SVM. They achieved an accuracy of 60.1% using quadratic SVM. However, they used principal component analysis (PCA) to reduce the dimensionality of the wavelet features. We found that the continuous wavelet transform features are highly non-linear. Therefore PCA is not a suitable dimensionality reduction method. In this article, we deal with dimensionality reduction using an autoencoder [24]. Hamid et al. [25] used the prosodic, spectral, and wavelet features to classify Arabic speech emotions. Many researchers have also utilized critical bands for speech-related tasks like SER and speaker identification. In 1961, Eberhard Zwicker proposed a psychoacoustic scale called Bark Bands [26]. The center frequencies in the bark bands are based on the human perception of different frequency ranges. Lalitha et al. [27] used a combination of the mel-scale and the bark-scale to perform SER and achieved encouraging results. Jiang et al. [28] used bark bands as a critical band division method and classified different emotions using a support vector machine. Their results showed that their proposed method performed better compared to MFCC features. Similarly, Fernandes et al. [29] also used bark bands as features and used an LSTM and a Bidirectional LSTM model for classification.

Several researchers have utilized wavelet packets also for SER. The wavelet packets is a generalization of multiresolution decomposition. It divides the frequency bands into several levels. Additionally, it decomposes the high-frequency portions also that are not subdivided in multiresolution analysis [30]. In 2020, Wang et al. [31] used the wavelet packet coefficients for SER and made a comparison with the MFCC features. They used a Sequential Floating Forward Search method for feature selection. Their experiments showed that the classifiers trained using the wavelets feature achieved better results than the MFCC features. Kishore et al. [32] used MFCC and Sub-Band Cepstral (SBC) features to classify emotions using a GMM. They computed SBC using the wavelet packet transform instead of the FFT. They reported that SBC features yielded a 70% accuracy and MFCC features yielded a 51% accuracy using the Surrey Audio-Visual Expressed Emotion (SAVEE) dataset.

Feng et al. [33] used wavelet packets and computed the energy of each sub-band to classify speech emotions. They classified these features using an LSTM based model. He et al. [34] proposed new features by computing energy entropy from wavelet packet frequency bands. They classified the features using the GMM algorithm. Huang et al. [35] proposed sub-band spectral centroid weighted wavelet packet cepstral coefficients for classifying emotions. Additionally, they fused the prosody and voice quality features with the wavelet packet features and classified them using a Deep belief network. Their results showed that this method performs SER efficiently even under noisy conditions. The use of wavelet packets proved useful in recognizing emotions under real-world noise conditions also [36].

As discussed above, researchers often use a combination of several features to increase the accuracy of an SER system. In this article, instead of using several kinds of time-frequency representations, we aim to use only one kind of robust representation that can capture distinguishable information. A comparison is made with other methods using other time-frequency representations where only one kind of representation is used instead of the fusion of different representations. Additionally, in many methods, the wavelets are not utilized efficiently. Therefore, we aim to revisit the wavelet transforms and explore their usage in SER. Fig. 3 illustrates the potential of using the wavelet multiresolution analysis for SER. The main contributions of this article are highlighted as follows:

1) We investigate the potential of the wavelet transform as features for SER by utilizing their multiresolution property. The continuous wavelet features are extracted within a suitable frequency range by analyzing the frequencies carrying the salient information.
2) We propose a method called WaDER to perform SER, which consists of two parts. Firstly, due to the high dimensionality of the wavelet features, an autoencoder is used to reduce the dimensionality of the wavelet features at each timestep. Secondly, a 1D CNN-LSTM based model performs classification using the latent space of the autoencoder.
3) We found that the wavelet transform features can efficiently distinguish between several emotions and perform an accurate classification compared to the other time-frequency representations. We achieved an unweighted accuracy (UA) of 81.45% and a weighted accuracy (WA) of 81.22% for speaker-dependent experiments using the RAVDESS dataset.

A list of nomenclature used throughout this article is provided in Table I.

TABLE I
LIST OF NOMENCLATURE USED IN THIS PAPER

Term	Meaning/ Referred To
ANN	Artificial Neural Network
AWGN	Additive White Gaussian Noise
BN	Batch Normalization
BN_{TD}	Time Distributed Batch Normalization
CFS	Correlation-based feature selection
CI	Confidence Interval
CMRN	Collective Multi-View Relation Network
CNN	Convolutional Neural Network
CWT	Continuous Wavelet Transform
DCNN	Deep Convolutional Neural Network
DGA	Density-Based Spatial Clustering of Application with Noise Genetic Algorithm
DSCNN	Deep Stride Convolutional Neural Network
ELU	Exponential Linear Unit
EMO-DB	Berlin Database of Emotional Speech
FFT	Fast Fourier Transform
GFCC	Gammatone Frequency Cepstral Coefficients
GMM	Gaussian Mixture Model
GResNet	Gated Residual Networks
HMM	Hidden Markov Model
IEMOCAP	Interactive Emotional Dyadic Motion Capture
LFLB	Local Feature Learning Block
LSTM	Long Short-term Memory
LPC	Linear Prediction Coefficients
MFCC	Mel-Frequency Cepstral Coefficients
MSE	Mean Squared Error
PCA	Principal Component Analysis
PSD	Power Spectral Density
RAVDESS	Ryerson Audio-Visual Database of Emotional Speech and Song
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Networks
SAVEE	Surrey Audio-Visual Expressed Emotion
SBC	Sub-Band Cepstral
SD	Speaker-dependent
SI	Speaker-independent
SNR	Signal-to-Noise Ratio
STFT	Short-Time Fourier Transform
SVM	Support Vector Machine
SER	Speech Emotion Recognition
UA	Unweighted Accuracy
VAD	Voice Activity Detection
WA	Weighted Accuracy

II. DATA

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) [37] is used in this research. RAVDESS dataset is an example of simulated dataset. It contains speech and song samples with different emotions in a North American accent. Only speech emotion samples are used in this work. The 16-bit audio files are sampled at 48 kHz and provided in the Waveform Audio File (WAV) format. The number of utterances is 1440. There are 24 professional actors and 60 trials per actor. Out of the 24 professional actors, 12 are female, and 12 are male. Each actor is asked to speak two different sentences with different emotions. The two sentences are "Dogs are sitting by the door" and "Kids are talking by the door".

The speech emotions categories include neutral, calm, happy, sad, angry, fearful, disgust, and surprised. Apart from the neutral category, each category of emotions is expressed at two levels of emotional intensity (normal and strong). There are 192 utterances of each emotion except the neutral state. The neutral state only has 96 utterances.

III. PROPOSED METHOD: WADER

A. Preprocessing

The audio files are sampled at 16 kHz to reduce the size of data without affecting the speech quality and intelligibility. Firstly, each audio clip's leading and trailing silence is trimmed because it contains no useful information. However, the silence occurring between words is not removed. It provides information about the speaking rate and helps distinguish between weak and strong emotions. For example, people tend to speak faster when they are angry; therefore, the duration of the silence will be less. However, when people speak with a calm emotion, the duration of silence can be longer. Secondly, each clip is normalized such that the mean is 0 and the standard deviation is 1.

B. Data Augmentation

Since there are fewer samples in the RAVDESS dataset, most SER algorithms tend to overfit. Therefore, data augmentation is performed to generate new samples and make the SER system more robust to noise. Additive White Gaussian Noise (AWGN) is the most widely used noise addition method. It can model the random processes that naturally occur in nature. Therefore, new samples are augmented from each trimmed audio sample using AWGN. The AWGN can be represented by (1) and (2). The noise is added with a Signal-to-Noise Ratio (SNR) between 15 dB and 30 dB.

$$Z_t \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

where t is a discrete timestep, σ^2 is the variance of the noise, and Z is the noise that is drawn from a normal distribution.

$$Y_t = X_t + Z_t \tag{2}$$

where t is a discrete timestep, Y is the output/ augmented signal, and X is the input signal.

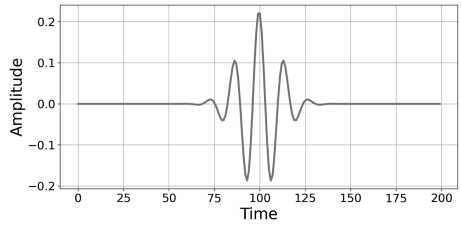


Fig. 1. Morlet mother wavelet.

C. Wavelet Feature Extraction

After data augmentation, each audio sample's continuous wavelet transform (CWT) is computed. The CWT of a signal is represented by (3).

$$\mathbf{W}(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} \mathbf{f}(t) \tilde{\psi} \left(\frac{t-b}{a} \right) dt \tag{3}$$

where a is the scale parameter, b is the translation parameter, $\psi(t)$ is the mother wavelet, $\tilde{\psi(t)}$ is the complex conjugate of the mother wavelet, and t is time. The scale and translation parameters in the CWT must be discretized to implement the algorithm. Different wavelet frequencies, F_a, are given by $\frac{F_c}{a\delta}$, where δ is the sampling period, and F_c is the central wavelet frequency which is set to 1 Hz. If the sampling period is $\frac{1}{16000}$ seconds, the scales 1, 2, and 3 correspond to 16 kHz, 8 kHz, and 5.33 kHz. Different discrete scales, a, are obtained by setting their values to positive integers, $a \in \{1, 2, 3, \ldots N\}$. N is chosen such that the frequency corresponding to the scale N is more than 20 Hz because the lower limit of human hearing is 20 Hz. Similarly, the translation values, b, in the CWT are discretized to positive integer values, which are the timesteps. $b \in \{1, 2, 3, \ldots, T\}$, where T is the total number of timesteps in the signal.

As parameters a and b change, different wavelets can be generated from the mother wavelet, which are called daughter wavelets. There are several kinds of mother wavelets like Daubechies, Mexican Hat, Symlet, Ricker, Haar, Morlet, etc. Different kinds of wavelets are used for different tasks. Morlet wavelet is well suited for speech and image processing tasks because it is closely related to human perception of hearing and vision. Over the past decade, it has been used for Voice Activity Detection (VAD), speaker recognition, and speech emotion recognition (SER).

The real-valued Morlet wavelet is used in this article and is shown in Fig. 1. The real-valued Morlet wavelet is given by (4).

$$\psi(t) = \cos(\xi t)\, e^{\frac{-t^2}{2\sigma^2}} \tag{4}$$

where σ is the width of the Gaussian, ξ controls the time and frequency resolutions trade-off, and t is time. The values of σ^2 and ξ are usually set to 1 and 5, respectively [23].

As the scale parameter increases, the daughter wavelet dilates/expands and captures lower frequencies. If the scale decreases, the daughter wavelet captures higher frequencies. If the mother wavelet is dilated by a factor of 2, it signifies that the frequency content is shifted by an octave. The number of octaves determines the number of frequencies being investigated.

For our experiments, the absolute values of the CWT features are taken. Now the crucial part is the selection of frequencies or scales. The human ear can hear frequencies between 20 Hz and 20,000 Hz. However, most speech lies between the 20 Hz and 4000 Hz range [38]. Therefore, selecting frequencies outside this range is not helpful. Additionally, to observe the difference in the distribution of the frequencies present in the male and female voices, the following approach is used:

- Firstly, each audio clip is standardized in the time domain (the mean is 0 and the standard deviation is 1).
- Secondly, the spectrogram of each audio clip is computed.
- Thirdly, all the frequencies are weighted by their amplitudes, and a histogram is computed as shown in Fig. 2. This process is repeated for both males and females separately.

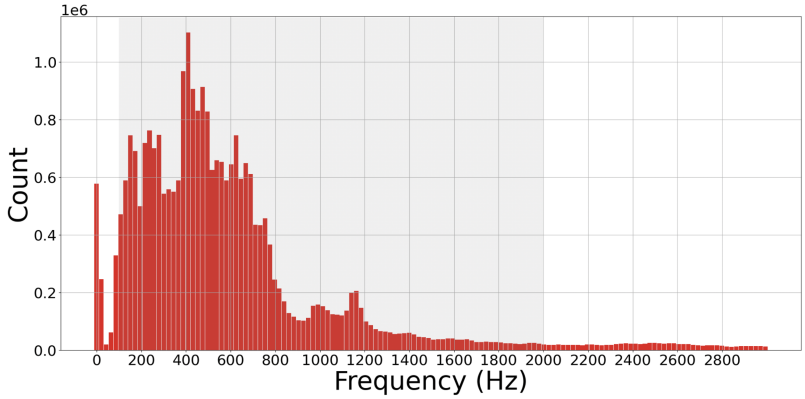
In the histograms, the frequencies between 80 Hz and 2000 Hz show a distinct pattern for males and females. Therefore, 125 frequencies in the range 80 Hz and 2000 Hz are extracted using their corresponding closest scales.

The CWT of all the emotions is shown in Fig. 3. The CWT of several emotions looks distinguishable. In Fig. 3, it can be seen that the CWT features of the "neutral", "calm", and "sad" emotions look similar. They have higher amplitudes in both lower and higher scales. However, the CWT features of emotions like "angry" and "disgust" show a higher amplitude in the lower scales (higher frequencies) only. It is due to a higher pitch when speaking loudly or angrily. The CWT of some emotions show a similar pattern, and it is difficult to differentiate between them visually. However, deep learning models should be able to uncover the complex underlying patterns from these CWT features to classify them accurately.

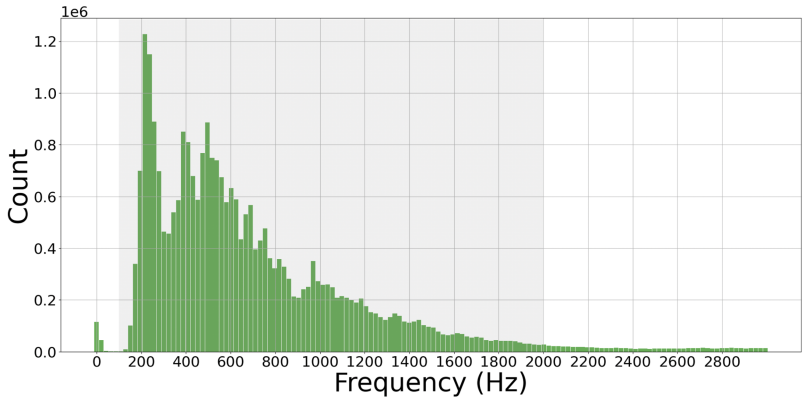
D. Fixed Size Segments Generation

Now we have the CWT features of all the audio samples. However, the duration of the audio samples is variable (between 3 and 5 seconds). To generate fixed-sized segments, one-second long segments are extracted from each sample with an overlap of 60%. The reason behind choosing one-second long segments is to capture multiple words and the pauses between them. Usually, some words are spoken in a more neutral manner than others. When we consider multiple words, we can estimate the emotion in that segment more accurately. Additionally, choosing long segments ensures that the segment's target emotion will not differ significantly from the whole utterance. The pauses between the words indicate how two words are connected, which helps distinguish between weaker (calm, neutral, sad) and stronger (angry, disgust, happy, surprised) emotions.

The ground truth label of the original utterance is assigned to its segments also. However, some portions of speech can carry a different (mostly neutral) emotion than the whole utterance. To address this issue, some methods dynamically generate pseudo labels for each segment [39]. However, in this article, the original utterance's label is also assigned to all its segments. During testing, the majority vote of the prediction of each segment is assigned to the whole utterance. Since the sampling rate is 16 kHz, one-second long segment corresponds to 16000 samples. Therefore, the dimensionality of each segment is $(16000, 125)$, where



(a) A histogram of male audio samples' frequencies weighted by their amplitudes.



(b) A histogram of female audio samples' frequencies weighted by their amplitudes.

Fig. 2. A histogram of male and female audio samples' frequencies weighted by their amplitudes using the RAVDESS dataset is shown in (a) and (b), respectively. Each audio sample is standardized in the time domain before computing its spectrogram. The histogram shows the difference in the distribution of frequencies present in the male and female voices. Most speech lies between the 20 Hz and 4000 Hz range. However, the region between 80 Hz and 2000 Hz (shown in gray) specifically shows the difference in the distribution of frequencies; therefore, the wavelet features in this frequency range are extracted.

the number of timesteps is 16000, and the number of scales is 125.

Currently, one major issue with the CWT features is the requirement of large-sized arrays, which makes it difficult to load the entire data during training. Therefore, the temporal resolution is decreased by a factor of 4 by treating the CWT features like images using inter-area interpolation.

Now, the dimensionality of each segment is $(4000, 125)$, where the number of timesteps is 4000, and the number of scales is 125. All the CWT features are standardized to have a mean of 0 and a standard deviation of 1.

E. Feature Compression

Currently, the number of features is significantly high at each timestep. Due to high dimensionality and fewer samples, the deep learning model is prone to overfitting. Therefore, a dimensionality reduction technique is applied.

Firstly, the most popular and simple dimensionality reduction technique, PCA, is explored. Currently, the CWT features form a $(N \times 4000 \times 125)$ matrix, where N is the number of samples (segments). The CWT features are reshaped to $(N \times 4000 * 125)$ to apply PCA.

A scatter plot of the first two principal components is shown in Fig. 4. It is evident from the scatter plot that the data is highly non-linear. But the PCA performs a linear transformation. Therefore, PCA is not a suitable dimensionality reduction method in this case.

Hence, the autoencoder is chosen to perform dimensionality reduction as it can also model non-linear data.

F. Deep Learning Architecture

The proposed deep learning architecture consists of an autoencoder and a classifier module. The autoencoder is used to reduce the dimensionality of features while keeping the number of timesteps the same. The classifier takes the latent space as input and classifies different emotions. The deep learning architecture is shown in Fig. 5.

Let the input CWT features be X, where $X \in \mathbb{R}^{T \times D}$, where T is the number of timesteps and D is the dimensionality of features. The encoder and decoder are represented by $e(.)$ and $d(.)$, respectively. The latent space, z, is represented by (5).

$$\mathbf{z} = e(\mathbf{X}) \tag{5}$$

where $z \in \mathbb{R}^{T \times D_1}$, and $D_1 < D$.

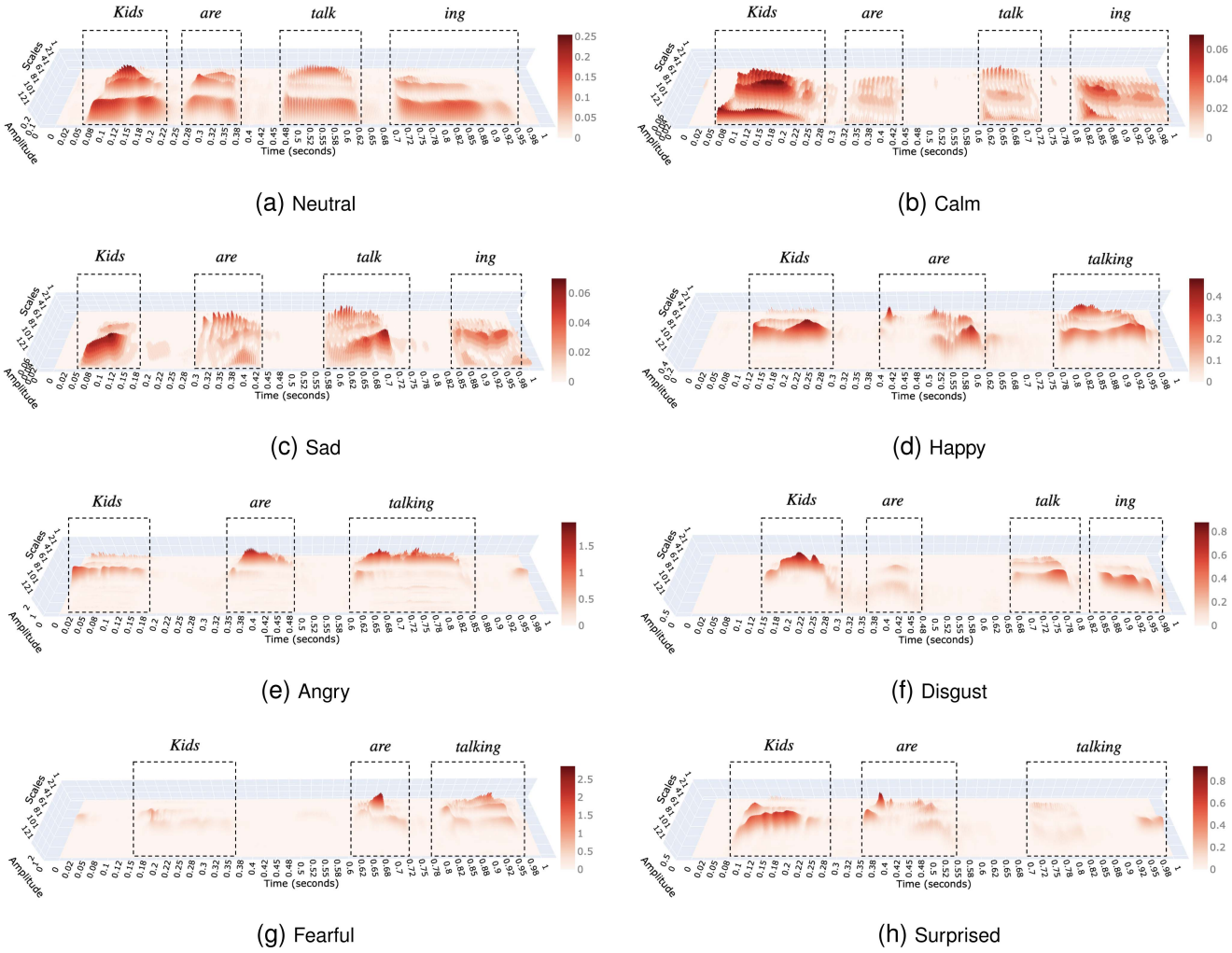


Fig. 3. The standardized Continuous Wavelet Transform (CWT) features of different emotions. A sample of each emotion is taken from the RAVDESS dataset. The sentence spoken in all the samples is "Kids are talking by the door". For convenience, the features from a 1 s long clip are extracted from each sample. The sampling rate is 16 kHz. Therefore, the number of timesteps shown is $16000 \times 1 = 16000$. In each emotion, only the words "Kids are talking" are uttered during the 1 s duration, which are annotated. The x-axis shows the time in seconds. The y-axis shows 125 scales, and the z-axis shows the amplitude. The scales are inversely proportional to frequencies. Note that the color axes vary by the plot to show the differences in amplitude per emotion.

The reconstructed CWT features, X', are described by (6).

$$\mathbf{X}' = d(\mathbf{z}) \tag{6}$$

where $X' \in \mathbb{R}^{T \times D}$.

The encoder and decoder use the time-distributed fully connected and time-distributed Batch Normalization layers. The time-distributed operation applies a specific layer to every timestep. This is done because the CWT features are viewed as a multivariate time series instead of a standard image here. Additionally, the temporal resolution is kept the same in the autoencoder. Only the dimensionality of the features at each timestep is reduced. Therefore, the feature maps are computed at every timestep using time-distributed layers.

The reconstruction loss, L_a, to train the autoencoder is given by (7).

$$L_a = \sum_{k=1}^{K} \left(||X - X'||^2 \right) \tag{7}$$

where K is the number of samples in the training data.

In our experiments, the values of D, D_1, and T are 125, 8, and 4000, respectively.

The classifier takes the latent space, z, as input. The classifier utilizes 1D CNNs to extract the spatial features at each timestep. Then, the LSTM layer is used to extract the long-term temporal dependencies as the input sequence length is 4000.

To extract the spatial dependencies across the scales at each timestep, three 1D convolutional layers, followed by the exponential linear unit (ELU) activation, Time-Distributed Batch Normalization, and MaxPooling layers, are used. The purpose of each layer is explained below:

1) *1D CNN:* 1D convolutional layers are applied to extract the spatial dependencies from the compressed frequency information in the latent space at each timestep. Instead of 2D CNN layers, 1D CNN layers are used because 1D CNNs require less memory during processing and are less computationally expensive. Moreover, the CWT

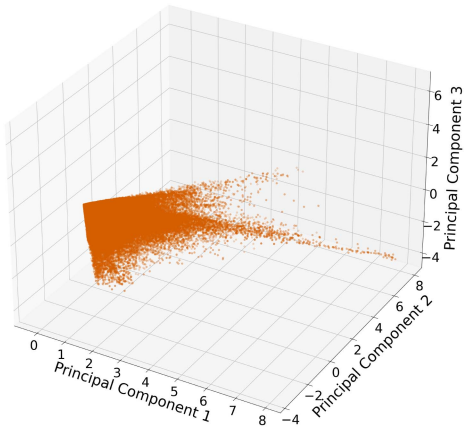


Fig. 4. The scatter plot of the first two principal components. The PCA is applied on the Continuous Wavelet Transform (CWT) features. It can be clearly observed that the features are non-linear as there is a significant spread in multiple directions.

features are considered here as a multivariate time series. Therefore, 1D CNN layers are utilized to learn the local features at each timestep.

2) *ELU activation:* The ELU activation is similar to ReLU activation but it can produce negative outputs [40]. The ELU activation is represented by (8).

$$ELU(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha(e^x - 1) & \text{if } x <= 0 \end{cases} \tag{8}$$

where α is the hyperparameter that adjusts the saturation for negative input values.

The ELU activation alleviates the effect of the vanishing gradient problem. Additionally, the ELU activation produces negative values, which pushes the mean of the activations closer to zero and results in faster training. Clevert et al. [40] showed that ELU activations lead to better generalization performance.

3) *Batch Normalization:* The distribution of the inputs of layers keeps changing in the neural network as the parameters of the previous layers change, which leads to slower training. This problem is termed "*Internal Covariate Shift*". Batch Normalization [41] adjusts the means and variances of layer inputs by normalizing them. The Batch Normalization layer reduces the dependency of gradients on initial parameters and makes the training faster by allowing higher learning rates [41], [42].

The time-distributed Batch Normalization layer applies the Batch Normalization at every timestep separately.

4) *Max Pooling:* 1D Max Pooling is used to reduce the temporal resolution of the CWT features by taking the maximum of a pooling region.

Let the input to the 1D CNN layer be a time series $X(t)$. The input $X(t)$ is convolved with a kernel $w(t)$ of size l to obtain the output $O(t)$, which is described using (9).

$$\mathbf{O}(t) = \mathbf{X}(t) * \mathbf{w}(t) = \sum_{k=-l}^{l} \mathbf{X}(k) \cdot \mathbf{w}(t-k) \tag{9}$$

The weights of the kernel $w(t)$ are initialized using Xavier normal initialization. Then, the output of the CNN layer can be represented using (10).

$$\mathbf{O}_i^l = b_i^l + \sum_k \mathbf{O}_k^{l-1} * \mathbf{w}_k^l \tag{10}$$

where O_i^l is the i^{th} output feature at l^{th} layer, O_k^{l-1} is the k^{th} input feature at the $(l-1)^{th}$ layer, w_k denotes the convolution kernel at the k^{th} index, and b_i^l is the bias term for the i^{th} output feature at the l^{th} layer.

The ELU activation function, $\sigma(.)$, is applied on the convolution output, $\mathbf{O}_i^l$. A time-distributed Batch Normalization layer, BN_{TD}, is applied to normalize the output of the activation function at each timestep, which is represented by (11). The α parameter in ELU activation is set to 1 (default value).

$$\mathbf{a}_i^l = BN_{TD}\left(\sigma\left(\mathbf{O}_i^l\right)\right) \tag{11}$$

Now, the outputs are passed into a MaxPooling layer which is shown in (12).

$$\mathbf{h}_i^l = \max_{\forall u \in \Omega_i} \mathbf{a}_u^l \tag{12}$$

where Ω represents the pooling region with index i, a_u^l is the input feature of the l^{th} MaxPooling layer at index u, and h_i^l is the output feature of the l^{th} MaxPooling layer at index i.

After extracting the features using the 1D CNN layers, an LSTM layer is applied to extract the long-term contextual dependencies from the CWT features. LSTM acts as a global feature extractor. The output from the LSTM cell, h_t^l, can be expressed using (13) to (17) [43].

$$\mathbf{f}_t = \sigma_g\left(\mathbf{W}_f \mathbf{h}_t^{l-1} + \mathbf{U}_f \mathbf{h}_{t-1}^l + \mathbf{b}_f\right) \tag{13}$$

$$\mathbf{i}_t = \sigma_g\left(\mathbf{W}_i \mathbf{h}_t^{l-1} + \mathbf{U}_i \mathbf{h}_{t-1}^l + \mathbf{b}_i\right) \tag{14}$$

$$\mathbf{o}_t = \sigma_g\left(\mathbf{W}_o \mathbf{h}_t^{l-1} + \mathbf{U}_o \mathbf{h}_{t-1}^l + \mathbf{b}_o\right) \tag{15}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \left(\sigma_c\left(\mathbf{W}_c \mathbf{h}_t^{l-1} + \mathbf{U}_c \mathbf{h}_{t-1}^l + \mathbf{b}_c\right)\right) \tag{16}$$

$$\mathbf{h}_t^l = \mathbf{o}_t \odot \mathbf{c}_t \tag{17}$$

where the W, U, and b terms denote the neural network weight matrices, σ_g is the logistic sigmoid function. i, o, and f are the input, output, and forget gates, respectively. i_t, o_t, and f_t are the gate vectors, c is the cell state, the operator $\odot$ represents the element-wise product of the vectors, σ_c is the hyperbolic tangent function, and l and $l-1$ denote the index of input and output features.

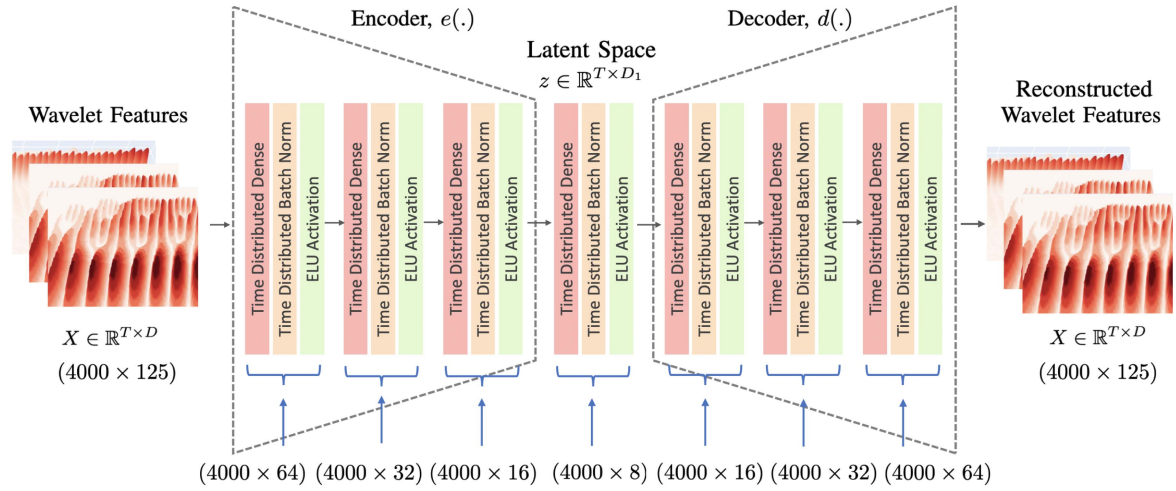
Note that our experiments use no activation on the LSTM layer's output (17).

The output of the LSTM layer is finally passed through fully connected layers, and a softmax activation is used on the final layer. The softmax output, $\hat{y}$ is represented by (18) and (19).

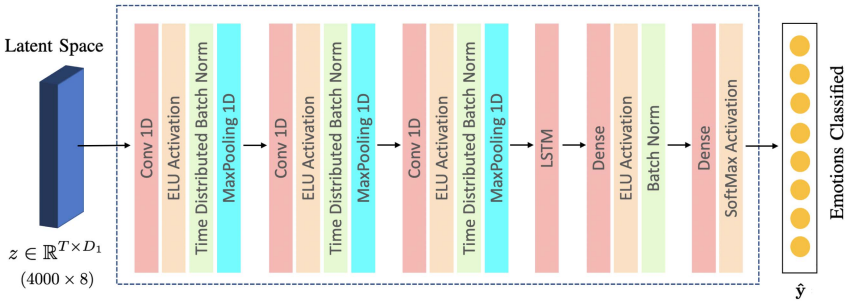
$$\mathbf{h}_1 = BN\left(\sigma_e\left(\mathbf{W}_1 \mathbf{h}^l + \mathbf{b}_1\right)\right) \tag{18}$$

$$\hat{\mathbf{y}} = \sigma_s\left(\mathbf{W}_2 \mathbf{h}_1 + \mathbf{b}_2\right) \tag{19}$$

where h^l is the output of the LSTM layer, W_1 and W_2 are the neural network weight matrices, BN is the Batch Normalization layer, σ_e is the ELU function, σ_s is the softmax function, h_1 is



(a) Autoencoder Architecture. The shape of each layer's output is represented in the format (x × y), where x is the number of timesteps and y is the dimensionality of the feature maps.



(b) Classifier Architecture

Fig. 5. The deep learning model consists of two parts: (a) The Autoencoder architecture. It reduces the dimensionality of the Continuous Wavelet Transform (CWT) features. The number of timesteps is kept the same, and only the number of features is reduced. (b) The Classifier model. The latent space of the autoencoder is used to classify all eight emotions from the RAVDESS dataset. T is the number of timesteps. D and D_1 are the dimensionality of CWT features and latent space, respectively.

the output of the first fully connected layer, and b_1 and b_2 are the bias matrices.

Using the softmax probabilities, the predicted class, y_{class}, is given by (20).

$$\mathbf{y}_{\text{class}} = \arg\max_i \hat{\mathbf{y}}_i \tag{20}$$

where $\hat{y}_i$ is the probability of the i^{th} class.

The classifier is trained using the categorical cross-entropy loss, L_c, which is represented by (21).

$$\mathbf{L}_c = \sum_{k=1}^{K} \mathbf{y}_k \log(\hat{\mathbf{y}}_k) \tag{21}$$

where K is the number of samples in the training data, y_k is the ground truth of sample k, and $\hat{y_k}$ is the prediction of the sample k.

After predicting the labels of segments, the majority vote, y_{vote}, of the labels of all the segments is assigned to the whole utterance.

The pseudo-code of the entire method is presented in Algorithm 1.

IV. Experiments

A. Evaluation Metric

The most widely used evaluation metrics for SER are weighted accuracy (WA) and unweighted accuracy (UA). WA is the average accuracy of all the samples which can be computed using (22).

$$WA = \frac{\sum_{k=1}^{K} n_k}{\sum_{k=1}^{K} N_k} \tag{22}$$

where k is the number of classes, n_k is the number of correctly classified samples in class k, and N_k is the total number of samples in class k.

UA is each class's average accuracy, which can be computed using (23). When the class distribution is skewed, WA is not a reliable metric. Therefore, for unbalanced classes, UA is used primarily.

$$UA = \frac{\sum_{k=1}^{K} n_k/N_k}{K} \tag{23}$$

Algorithm 1: Speech Emotion Recognition using the Continuous Wavelet Transform (CWT) Features.

Input: $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_N\}$: N standardized audio clips from the RAVDESS dataset after trimming the leading and trailing silence.

Output: $\mathbf{P} = \{\}$: The predicted labels of the audio clips.

1: $X_{all} =$ Training CWT features
2: **for** $i = 1$ to N **do**
3: Read the audio clip, Y_i. Augment two new samples, Y_i^1 and Y_i^2 by adding Additive White Gaussian Noise with an SNR between 15 dB and 30 dB.
4: Extract the CWT features from all three samples.
5: Divide the CWT features into fixed sized segments.
6: Decrease the temporal resolution of CWT features, $\mathbb{R}^{16000\times125} \rightarrow \mathbb{R}^{4000\times125}$.
7: Append the CWT features into the array, X_{all}.
8: **end for**

X_{all} is a $(N_1 \times T \times D)$ matrix, where N_1 is the number of segments, T is the number of timesteps, and D is the dimensionality of CWT features.

9: Standardize the features array, X_{all}.
10: Train the autoencoder using features X_{all}.
11: Train an ensemble of seven classification models. Use the latent space, $\mathbf{z} \in \mathbb{R}^{T\times D_1}$, of the autoencoder as the input features for the classification model. D_1 is the dimensionality of the latent space and $D_1 < D$.
12: *Segment-level prediction:* Make prediction, y_{class}, for each segment using the ensemble of models.
13: *Utterance-level prediction:* Compute the majority vote, y_{vote}, of the predicted labels of segments to make prediction at the utterance level
14: Append y_{vote} into the array P.
15: **return P**

For the experiments, the value of T, D, and D1 are 4000, 125, and 8, respectively.

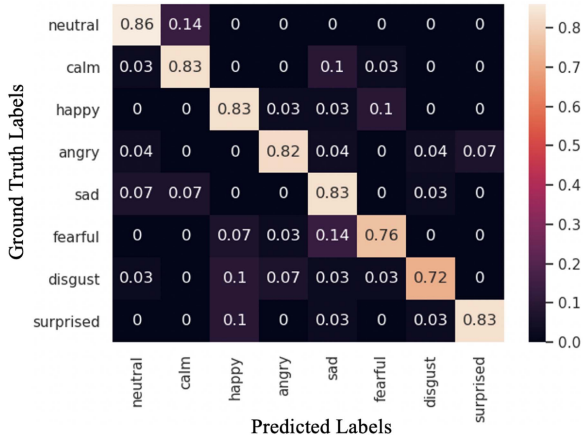


Fig. 6. Confusion matrix of the eight emotions classification from the RAVDESS dataset. The Continuous Wavelet Transform (CWT) features are used to classify different emotions. The deep learning model yielded a mean UA (%) and WA (%) of $81.45 \pm 1.19\%$ and $81.22 \pm 1.31\%$, respectively. The highest UA (%) and WA (%) achieved are 83.4% and 83.7%, respectively. A speaker-dependent (SD) speech emotion recognition is performed here.

TABLE II
THE MEAN SQUARED ERROR, AND THE R^2 SCORE OF THE RECONSTRUCTED CONTINUOUS WAVELET TRANSFORM (CWT) FEATURES FROM THE AUTOENCODER. THE RAVDESS DATASET IS USED HERE

Emotion	Reconstruction using Autoencoder	
	MSE	$\mathbf{R^2}$ Score ($\pm$0.02)
Neutral	0.038 ± 0.015	0.90
Calm	0.026 ± 0.008	0.93
Happy	0.029 ± 0.012	0.92
Sad	0.021 ± 0.010	0.95
Angry	0.023 ± 0.005	0.86
Fearful	0.018 ± 0.010	0.94
Disgust	0.023 ± 0.012	0.94
Surprised	0.020 ± 0.014	0.90
Mean	$\mathbf{0.024 \pm 0.010}$	**0.92**
* Data approx. range $\in (-0.4, 89)$		

where k is the number of classes, n_k is the number of correctly classified samples in class k, and N_k is the total number of samples in class k.

B. Experimental Setup

A speaker-dependent (SD) speech emotion recognition is performed here. A Monte Carlo experiment is conducted to get a robust estimate of performance. The Monte Carlo simulation involves seven experiments overall. At the beginning of each experiment, data is randomly split into training and testing. 85% of the samples from each class are used for training, and 15% of the samples from each class are used for testing.

Note that the same training data is used for both autoencoder and classifier models to prevent data leakage.

1) *Autoencoder:* The autoencoder is trained using 5-fold cross-validation. In each fold, 10% of the data is used for validation. The model is trained for 40 epochs in each fold. The batch size is set to 128, and the learning rate is set to 0.0001. The autoencoder model is trained using the Adam optimizer and the mean squared error (MSE) loss function. The mean UA (%) and WA (%) from the Monte Carlo experiment are reported. The mean squared error and R^2-score of the reconstructed CWT features are shown in Table II.
2) *Classifier:* The classification deep neural network is trained using 8-fold cross-validation. In each fold, 10% of the data is used for validation. The model is trained for 25 epochs in each fold. The batch size is set to 64, and the learning rate is set to 0.0001. The classifier model is trained using the Adam optimizer and categorical cross-entropy loss function. An ensemble of 7 classification models is used to make robust predictions. All the models have the same architecture but are initialized with different weights.

The autoencoder and the classifier are trained using the CWT features of segments and not the whole utterance. Therefore, after predicting the labels of segments, a majority vote of the predicted labels of all the segments is taken. The label of the majority of the segments is assigned to the whole utterance. The tie is broken by selecting a random sample. In the autoencoder

TABLE III
COMPARISON OF THE PROPOSED METHOD WITH STATE-OF-THE-ART SER METHODS

Method	Speaker Dependent (SD) Mean Accuracy	
	WA (%)	UA (%)
Head Fusion [15]	77.8	77.4
BiLSTM + Capsule Routing [44]	–	69.4
DCNN [45]	–	71.6
DSCNN + Raw spectrogams [46]	68.0	61.0
DSCNN + Cleaned spectrogams [46]	80.0	79.0
Spectrograms + GResNets [47]	64.5	64.6
WaDER *(Ours)*	**81.2**	**81.4**

A speaker-dependent (SD) speech emotion recognition is performed here. The mean weighted and unweighted accuracies are reported.

and the classifier models, experiments are conducted by switching the order of the batch normalization and ELU activation layers, and a similar performance is observed.

C. Results

The mean UA (%) and WA (%) achieved from the Monte Carlo experiment are $81.45 \pm 1.19\%$ and $81.22 \pm 1.31\%$, respectively. The confusion matrix of one of the experiments is shown in Fig. 6. The 95% confidence interval (CI) of UA (%) is $[78.80, 83.11]\%$. The 95% CI of WA (%) is $[79.38, 83.05]\%$. The highest UA and WA achieved are 83.4% and 83.7%, respectively.

D. Comparison and Analysis

To validate the effectiveness of the proposed method, WaDER, a comparison is made with the state-of-the-art methods in terms of speaker-dependent unweighted and weighted accuracy. The comparison is made using the mean accuracy reported by different methods, [15, Tab. VII], and [48, Tab. IX]. The proposed method outperforms all the other methods in terms of unweighted and weighted accuracy. The best unweighted and weighted accuracies achieved by our method are 83.4% and 83.7%, respectively. Our method is similar to the work done by Aghajani et al. [49]. One difference is that we have reduced the dimensionality of the wavelet features using an autoencoder and chosen the wavelet scales carefully. The classifier feature extraction blocks are slightly similar to the local feature learning block (LFLB) by Zhao et al. [9]. Some methods show better results than our method. However, a direct comparison with those methods is not possible because either they are classifying only a few classes from a benchmark dataset or they use a combination of various features in their method [48]. These methods focus on increasing SER accuracy by combining different kinds of features. However, this work aims to explore the potential of the wavelet transform as features for SER. Additionally, instead of time-frequency domain features, some methods [50], [51] use embeddings from a pretrained model as features to train their SER model. Therefore, a comparison is not made with such methods. However, Farooq et al. [51] achieved a mean weighted accuracy of 81.3%, which is only 0.1% more than our method's mean weighted accuracy. Farooq et al. [51] used the features from a pretrained Alexnet, which resulted in a better performance. The 95% confidence interval of our model's weighted and unweighted accuracies also overlaps with the accuracies reported by Kwon et al. [46].

The confusion matrix in Fig. 6 shows that the proposed method shows high accuracy for all the emotions. However, some confusion exists between "neutral" and "calm" samples. It is quite challenging to differentiate between "neutral" and "calm" samples as they both have similar pitch and speaking rates. It is difficult for human listeners also to distinguish between these two classes with a 100% accuracy. Therefore, several methods sometimes merge these two classes into a single class for classification because of their high similarity. Similarly, there are some ambiguities between "calm" and "sad" samples as well. Some strong emotions like "angry", "disgust", "surprised", and "fearful" also have slight confusion as they all possess higher amplitudes in higher frequencies.

One key point was reducing the dimensionality of the continuous wavelet transform features. The autoencoder compresses the wavelet features by a factor of 15.6. It can be seen from Table II that the autoencoder is able to reconstruct the data from the latent space efficiently. The optimal size of latent space was found to be $(4000, 8)$ (the original size was $(4000, 125)$) after experimenting with different latent space sizes. If the size of the latent space is further decreased, more information is lost, and the wavelet features are reconstructed with a higher loss. On the other hand, if the size of the latent space is increased, the classifier begins to overfit. The primary reason behind choosing an ensemble of seven models is to avoid overfitting. Since the RAVDESS dataset contains fewer samples, new samples are augmented by adding additive white gaussian noise with an SNR between 15 dB and 30 dB, which results in a robust performance. However, the model is still very prone to overfitting. If the model is trained for more epochs, the model immediately begins to overfit. Therefore, the training is stopped as soon as the model begins to overfit. However, the proposed method still outperforms the other methods in terms of weighted and unweighted accuracy.

V. CONCLUSION

This article uses the continuous wavelet transforms as features to perform speech emotion recognition. The proposed method, WaDER, firstly uses an autoencoder to reduce the dimensionality of the wavelet features. Secondly, the latent space is used to perform classification using an ensemble of seven deep neural networks. The experiments are conducted on the RAVDESS

dataset. We observed that the continuous wavelet transform features are able to distinguish between several emotions and perform an accurate and robust classification. We showed the potential of utilizing the multi-resolution property of the wavelets to classify emotions. However, the current methodology still requires some improvements. Firstly, extracting features from frequencies that carry the most discriminative emotional information could improve the SER performance. A channel-wise attention mechanism that can extract the salient features from the frequencies at each timestep can be used to achieve this. Secondly, a new strategy is required to overcome the severe overfitting problem. Thirdly, we need to extend the model to the speaker-independent (SI) scenarios and evaluate the performance. Our future work will focus on overcoming these challenges and developing a more efficient SER architecture.

REFERENCES

[1] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *J. Acoust. Soc. America*, vol. 8, no. 3, pp. 185–190, 1937.

[2] A. Nogueiras et al., "Speech emotion recognition using hidden Markov models," in *Proc. 7th Eur. Conf. Speech Commun. Tech.*, 2001.

[3] B. Schuller, G. Rigoll, and M. Lang, "Hidden Markov model-based speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, pp. II–1.

[4] T. Seehapoch and S. Wongthanavasu, "Speech emotion recognition using support vector machines," in *Proc. IEEE 5th Int. Conf. Knowl. Smart Technol.*, 2013, pp. 86–91.

[5] Y. Chavhan, M. Dhore, and P. Yesaware, "Speech emotion recognition using support vector machine," *Int. J. Comput. Appl.*, vol. 1, no. 20, pp. 6–9, 2010.

[6] L. Sun, S. Fu, and F. Wang, "Decision tree SVM model with fisher feature selection for speech emotion recognition," *EURASIP J. Audio, Speech, Music Process.*, vol. 2019, no. 1, pp. 1–14, 2019.

[7] C. Yogesh et al., "Bispectral features and mean shift clustering for stress and emotion recognition from natural speech," *Comput. Elect. Eng.*, vol. 62, pp. 676–691, 2017.

[8] S. Zhang, S. Zhang, T. Huang, and W. Gao, "Speech emotion recognition using deep convolutional neural network and discriminant temporal pyramid matching," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1576–1590, Jun. 2018.

[9] J. Zhao, X. Mao, and L. Chen, "Speech emotion recognition using deep 1D & 2D CNN LSTM networks," *Biomed. Signal Process. Control*, vol. 47, pp. 312–323, 2019.

[10] S. Khorram, Z. Aldeneh, D. Dimitriadis, M. McInnis, and E. M. Provost, "Capturing long-term temporal dependencies with convolutional networks for continuous emotion recognition," in *Proc. Interspeech*, 2017, pp. 1253–1257.

[11] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. 4th Int. Conf. Learn. Representations*, San Juan, Puerto Rico, May 24, 2016. [Online]. Available: https://doi.org/10.48550/arxiv.1511.07122

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*, pp. 1–15, doi: 10.48550/arXiv.1409.0473.

[13] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, "Speech emotion classification using attention-based LSTM," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 11, pp. 1675–1685, Nov. 2019.

[14] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 2227–2231.

[15] M. Xu, F. Zhang, and W. Zhang, "Head fusion: Improving the accuracy and robustness of speech emotion recognition on the IEMOCAP and RAVDESS dataset," *IEEE Access*, vol. 9, pp. 74539–74549, 2021.

[16] Y. Yu and Y.-J. Kim, "Attention-LSTM-attention model for speech emotion recognition and analysis of IEMOCAP database," *Electronics*, vol. 9, no. 5, 2020, Art. no. 713.

[17] S. K. Pandey, H. S. Shekhawat, and S. Prasanna, "Emotion recognition from raw speech using wavenet," in *Proc. IEEE Region Conf.*, 2019, pp. 1292–1297.

[18] A. J. Oxenham, "How we hear: The perception and neural coding of sound," *Annu. Rev. Psychol.*, vol. 69, pp. 27–50, 2018.

[19] J.-O. Strömberg, "A modified Franklin system and higher-order spline systems on Rn as unconditional bases for Hardy spaces," in *Fundamental Papers in Wavelet Theory*. Princeton, NJ, USA: Princeton Univ. Press, 2009, pp. 197–215.

[20] A. Grossmann and J. Morlet, "Decomposition of hardy functions into square integrable wavelets of constant shape," *SIAM J. Math. Anal.*, vol. 15, no. 4, pp. 723–736, 1984.

[21] A. Grossmann, J. Morlet, and T. Paul, "Transforms associated to square integrable group representations. I. General results," *J. Math. Phys.*, vol. 26, no. 10, pp. 2473–2479, 1985.

[22] H. Zhiyan and W. Jian, "Speech emotion recognition based on wavelet transform and improved HMM," in *Proc. IEEE 25th Chin. Control Decis. Conf.*, 2013, pp. 3156–3159.

[23] P. Shegokar and P. Sircar, "Continuous wavelet transform based speech emotion recognition," in *Proc. IEEE 10th Int. Conf. Signal Process. Commun. Syst.*, 2016, pp. 1–8.

[24] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ. San Diego La Jolla Inst. for Cognitive Science, San Diego, CA, USA, Tech. Rep., 1985.

[25] L. Abdel-Hamid, "Egyptian arabic speech emotion recognition using prosodic, spectral and wavelet features," *Speech Commun.*, vol. 122, pp. 19–30, 2020.

[26] H. Fastl and E. Zwicker, *Psychoacoustics: Facts and Models*, vol. 22. Berlin Heidelberg: Springer Science & Business Media, 2006.

[27] S. Lalitha and S. Tripathi, "Emotion detection using perceptual based speech features," in *Proc. IEEE Annu. India Conf.*, 2016, pp. 1–5.

[28] L. Jiang, P. Tan, J. Yang, X. Liu, and C. Wang, "Speech emotion recognition using emotion perception spectral feature," *Concurrency Comput.: Pract. Experience*, vol. 33, no. 11, 2021, Art. no. e5427.

[29] B. Fernandes and K. Mannepalli, "Speech emotion recognition using deep learning LSTM for Tamil language," *Pertanika J. Sci. Technol.*, vol. 29, no. 3, pp. 1915–1936, 2021.

[30] Y. Wu and W. Wu, "Analysis of wavelet decomposition properties of wind turbine signal," *Energy Rep.*, vol. 7, pp. 873–879, 2021.

[31] K. Wang, G. Su, L. Liu, and S. Wang, "Wavelet packet analysis for speaker-independent emotion recognition," *Neurocomputing*, vol. 398, pp. 257–264, 2020.

[32] K. K. Kishore and P. K. Satish, "Emotion recognition in speech using MFCC and wavelet features," in *Proc. IEEE 3rd Int. Adv. Comput. Conf.*, 2013, pp. 842–847.

[33] T. Feng and S. Yang, "Speech emotion recognition based on LSTM and MEL scale wavelet packet decomposition," in *Proc. Int. Conf. Algorithms, Comput. Artif. Intell.*, 2018, pp. 1–7.

[34] L. He, M. Lech, J. Zhang, X. Ren, and L. Deng, "Study of wavelet packet energy entropy for emotion classification in speech and glottal signals," *Proc. SPIE*, vol. 8878, pp. 581–586, 2013.

[35] Y. Huang, K. Tian, A. Wu, and G. Zhang, "Feature fusion methods research based on deep belief networks for speech emotion recognition under noise condition," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 5, pp. 1787–1798, 2019.

[36] J. C. Vásquez-Correa, N. García, J. R. Orozco-Arroyave, J. D. Arias-Londoño, J. F. Vargas-Bonilla, and E. Nöth, "Emotion recognition from speech under environmental noise conditions using wavelet decomposition," in *Proc. IEEE Int. Carnahan Conf. Secur. Technol.*, 2015, pp. 247–252.

[37] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north American english," *PLoS One*, vol. 13, no. 5, 2018, Art. no. e0196391.

[38] N. I. o. Health (US) and B. S. C. Study, " Information about hearing, communication, and understanding," in *NIH Curriculum Suppl. Ser. [Internet]*. National Institutes of Health (US), 2007. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK20366/

[39] S. Mao, P. Ching, and T. Lee, "Enhancing segment-based speech emotion recognition by iterative self-learning," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 123–134, 2022.

[40] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," *Comput. Sci.*, 2015.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[42] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?," in*Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 2488–2498.
[43] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," 2014, *arXiv:1402.1128*.
[44] M. A. Jalal, E. Loweimi, R. K. Moore, and T. Hain, "Learning temporal clusters using capsule routing for speech emotion recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 1701–1705.
[45] D. Issa, M. F. Demirci, and A. Yazici, "Speech emotion recognition with deep convolutional neural networks," *Biomed. Signal Process. Control*, vol. 59, 2020, Art. no. 101894.
[46] S. Kwon, "A CNN-assisted enhanced audio signal processing for speech emotion recognition," *Sensors*, vol. 20, no. 1, 2019, Art. no. 183.
[47] Y. Zeng, H. Mao, D. Peng, and Z. Yi, "Spectrogram based multi-task audio classification," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3705–3722, 2019.
[48] S. Kanwal and S. Asghar, "Speech emotion recognition using clustering based ga-optimized feature set," *IEEE Access*, vol. 9, pp. 125830–125842, 2021.
[49] K. Aghajani and I. E. P. Afrakoti, "Speech emotion recognition using scalogram based deep structure," *Int. J. Eng.*, vol. 33, no. 2, pp. 285–292, 2020.
[50] M. Sajjad et al., "Clustering-based speech emotion recognition by incorporating learned features and deep BILSTM," *IEEE Access*, vol. 8, pp. 79861–79875, 2020.
[51] M. Farooq, F. Hussain, N. K. Baloch, F. R. Raja, H. Yu, and Y. B. Zikria, "Impact of feature selection algorithm on speech emotion recognition using deep convolutional neural network," *Sensors*, vol. 20, no. 21, 2020, Art. no. 6008.

Aditya Dutt (Graduate Student Member, IEEE) received the M.S. degree in computer science in 2019 from the University of Florida, Gainesville, FL, USA, where he is currently working toward the Ph.D. degree in computer science with the Department of Computer and Information Science and Engineering.

His research interests include machine learning, metric learning, multimodal data fusion, speech analysis, and speech emotion recognition.

Paul Gader (Fellow, IEEE) received the Ph.D. degree in mathematics for image-processing-related research from the University of Florida, Gainesville, FL, USA, in 1986.

He is currently a Professor with the Department of Computer and Information Science and Engineering and the Engineering School of Sustainable Infrastructure and Environment, University of Florida. He performed his first research in image processing in 1984, working on algorithms for detecting bridges in forward-looking infrared imagery as a Summer Student Fellow with Eglin Air Force Base. He has been a leading figure in handwriting recognition and landmine detection. He led the development of a 5th-ranked handwritten character recognizer and a top-ranked handwritten word recognizer in two National Institute of Standards and Technology (NIST) competitions in the early 1990s. He has authored or coauthored more than 100 journals and more than 300 total papers and was an Associate Editor forIEEE GEOSCIENCE AND REMOTE SENSING LETTERS.

He has worked on a wide variety of theoretical and applied research problems, including fast computing with linear algebra, mathematical morphology, fuzzy sets, Bayesian methods, handwriting recognition, automatic target recognition, biomedical image analysis, landmine detection, human geography, and hyperspectral and light detection, and ranging image analysis projects.