electronics

MDPI

Article

A Deep Learning Approach for Speech Emotion Recognition Optimization Using Meta-Learning

Lara Toledo Cordeiro Ottoni [1,*], André Luiz Carvalho Ottoni [2] and Jés de Jesus Fiais Cerqueira [3]

[1] Graduate Program in Electrical Engineering, Federal University of Bahia, Salvador 40210-910, Brazil
[2] Technologic and Exact Center, Federal University of Recôncavo da Bahia, Cruz das Almas 44380-000, Brazil; andre.ottoni@ufrb.edu.br
[3] Department of Electrical and Computer Engineering, Federal University of Bahia, Salvador 40210-910, Brazil; jes@ufba.br
* Correspondence: lara.toledo@ufba.br

Abstract: Speech emotion recognition (SER) is widely applicable today, benefiting areas such as entertainment, robotics, and healthcare. This emotional understanding enhances user-machine interaction, making systems more responsive and providing more natural experiences. In robotics, SER is useful in home assistance devices, eldercare, and special education, facilitating effective communication. Additionally, in healthcare settings, it can monitor patients' emotional well-being. However, achieving high levels of accuracy is challenging and complicated by the need to select the best combination of machine learning algorithms, hyperparameters, datasets, data augmentation, and feature extraction methods. Therefore, this study aims to develop a deep learning approach for optimal SER configurations. It delves into the domains of optimizer settings, learning rates, data augmentation techniques, feature extraction methods, and neural architectures for the RAVDESS, TESS, SAVEE, and R+T+S (RAVDESS+TESS+SAVEE) datasets. After finding the best SER configurations, meta-learning is carried out, transferring the best configurations to two additional datasets, CREMA-D and R+T+S+C (RAVDESS+TESS+SAVEE+CREMA-D). The developed approach proved effective in finding the best configurations, achieving an accuracy of 97.01% for RAVDESS, 100% for TESS, 90.62% for SAVEE, and 97.37% for R+T+S. Furthermore, using meta-learning, the CREMA-D and R+T+S+C datasets achieved accuracies of 83.28% and 90.94%, respectively.

Keywords: speech emotion recognition; convolutional neural network; meta-learning; data augmentation

Citation: Ottoni, L.T.C.; Ottoni, A.L.C.; Cerqueira, J.d.J.F. A Deep Learning Approach for Speech Emotion Recognition Optimization Using Meta-Learning. *Electronics* **2023**, *12*, 4859. https://doi.org/10.3390/electronics12234859

Academic Editors: Zbigniew Leonowicz, Arkaitz Zubiaga and Michał Jasiński

Received: 30 October 2023
Revised: 26 November 2023
Accepted: 29 November 2023
Published: 1 December 2023

1. Introduction

Speech emotion recognition (SER) has been gaining increasing popularity in the growing field of human–computer interactions (HCIs) [1]. This is due to the significance of analyzing emotions expressed in speech to enhance the intelligence level of conversational robotics and HCI systems [2]. By interacting with individuals and understanding the emotions conveyed in speech, it is possible to provide higher-quality services and create a more intelligent, natural, and personalized human–computer interaction experience [3]. The significance of speech emotion recognition has significantly expanded across various sectors, encompassing applications in home automation, customer service, medical applications, and even entertainment [4]. In all these applications, effective communication between humans and computers/machines/robots requires understanding human intentions, which is often discerned through speech emotions [1,5].

In the early days of SER research, the primary focus was on probabilistic models such as Hidden Markov Models (HMMs) [6,7] and Gaussian Mixture Models (GMMs) [8,9]. Recently, with the advent of deep learning, the landscape of emotion recognition has significantly shifted towards neural-network-based approaches [4,10]. Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Deep Neural Networks (DNNs) now play a predominant role in advancing speech emotion recognition [11].

With the ongoing progress of deep learning techniques, specific challenges have arisen, including the need for readily available data to support new advancements and facilitate model comparisons [12]. In the field of SER, many small datasets are prevalent, leading to the merging of different databases and the application of data augmentation techniques, which play a pivotal role in enriching datasets. This, in turn, allows models to learn from a broad spectrum of emotional speech variations [4]. Another essential requirement is tuning various hyperparameter settings, such as the optimizer and learning rate [13]. In [14], it became evident that hyperparameter tuning significantly impacts the performance of deep learning models. Therefore, conducting tests to discover the optimal hyperparameters for each problem is crucial. Furthermore, a thorough investigation of various feature extraction techniques is essential in speech emotion recognition since the extracted features play a critical role in the quality and performance of models [15]. Last but not least, it is imperative to explore a variety of neural architectures, such as CNNs, RNNs, and hybrid architectures, to determine which ones deliver the best performance and generalization across different datasets and application scenarios. This comprehensive approach is essential for driving progress in the field of speech emotion recognition [3].

Notably, optimizing many combinations requires a long computational time before finding a good solution [16–18]. In this context, a recently highlighted approach that has attracted considerable attention is meta-learning (MtL). MtL involves building strategies that enable models to learn from diverse datasets and transfer that knowledge to related tasks, accelerating the learning process and improving generalization [19]. An example is the transfer of hyperparameters to similar datasets [20].

However, despite the considerable amount of work in SER, gaps in the existing literature studies are observed. The literature still lacks an approach based on deep learning that explores the combinations between optimizers, learning rates, data augmentation techniques, feature extraction, and the neural architecture. Most works present limited comparisons, focusing on specific aspects such as neural architectures or feature extraction techniques in isolation. Additionally, the absence of studies that employ meta-learning and hyperparameter optimization, such as optimizers and learning rates, is notable. These aspects serve as motivation for our paper.

Thus, this study aims to develop a deep learning approach that investigates the best configurations of optimizers, learning rates, data augmentation techniques, feature extraction methods, and neural architectures. To achieve this, the approach is applied to four datasets: RAVDESS, TESS, SAVEE, and R+T+S (RAVDESS+TESS+SAVEE). Once the optimal combination is identified, meta-learning is performed, i.e., transferring the best configuration to two additional databases: CREMA-D and R+T+S+C (RAVDESS+TESS+SAVEE+CREMA-D).

In summary, the main contributions of this study are:

- A deep learning approach is proposed that investigates the best configurations of optimizers, learning rates, data augmentation, feature extraction, and the neural architecture for different datasets.
- Meta-learning is performed, transferring the best configuration found for the optimizer, learning rate, data augmentation, feature extraction, and neural architecture to two other SER datasets.

This article is structured as follows: Section 2 reviews the research in speech emotion recognition that investigates the best SER configuration. Section 3 defines the complete approach for obtaining results for each dataset under study and for meta-learning. The results are presented and discussed in Section 4, while final considerations and conclusions are presented in Section 5.

2. Related Works

The field of SER research is highly dynamic and has seen many significant innovations and advances over time. Specifically, advanced deep learning (DL) techniques have brought substantial progress in this area. Implementing DL and DNNs generated the need to investigate and adjust settings for a better model performance.

Hyperparameter tuning, such as optimizers and learning rates, significantly improves the classifier [13]. Furthermore, with the scarcity of datasets in the SER field, data augmentation is a valuable alternative to compensate for the lack of data [15]. Furthermore, carefully selecting features extracted from audio plays a crucial role in the overall performance of the applied DL techniques [21]. Finally, the choice of neural architecture used to classify emotions expressed in speech is a determining factor [22]. These elements are fundamental and must be carefully considered, as they directly influence the classifier's performance, when aiming to achieve the best results.

In this context, we explore recent works investigating the optimal combination of SER configurations. These articles compare at least one SER parameter: the optimizer, the learning rate, feature extraction, data augmentation, the neural architecture, and meta-learning. In Table 1, it is possible to observe the papers and configurations investigated in their work.

Table 1. Related work in the SER area that investigates the best combinations of optimizers, learning rates, feature extraction, neural architectures, and meta-learning. Papers marked with a check indicate that they performed a comparison with the respective configuration.

Paper	Optimizer	Learning Rate	Feature Extraction	Data Augmentation	Architecture	Meta-Learning
[23]	-	-	-	✓	✓	-
[22]	-	-	-	✓	✓	-
[24]	-	-	-	✓	-	-
[25]	-	-	-	✓	✓	-
[26]	✓	-	✓	-	-	-
[27]	-	-	-	-	✓	-
[10]	-	-	✓	✓	✓	-
[28]	-	-	-	✓	✓	-
Proposed	✓	✓	✓	✓	✓	✓

In [23], the authors use the RAVDESS dataset and employ the MFCC technique to extract audio features. The article compares data augmentation, including adding noise and pitch shifting. Furthermore, the neural architecture is compared, comparing a CNN to a CNN+LSTM. The authors do not provide details on how the optimizer and learning rate were chosen or specify which hyperparameters were used.

In [22], the authors also compare the use and non-use of data augmentation and compare neural architectures. They used the RAVDESS, TESS, SAVEE, CREMA-D, and EMO-DB datasets. The data augmentation techniques compared include noise, pitch shifting, and stretching. The comparison was conducted between not using data augmentation and all these techniques together (noise+pitch+stretch). Additionally, the authors investigated neural architectures, comparing a CNN, CNN+LSTM, and CNN+GRU. They used the Adam optimizer with an adjustable learning rate without further investigations.

On the other hand, in [24], only data augmentation was compared. The applied data augmentation operations were noise, pitch, and stretching. A 1D CNN was used, and the techniques used to extract the features were MFCC, Mel, Chroma, and ZCR together. The hyperparameters employed consisted of the Adam optimizer with an adaptive learning rate.

In [25], the authors compare different types of data augmentation and different neural architectures. The data augmentation techniques compared include Gaussian Noise, SpecAugment, Room Impulse Response (RIR), and Tanh Distortion. The neural architectures compared are CNN2D, CNN+BiLSTM+Attention, and CNN+Transformer. The feature extraction information used by the models is the Mel spectrogram. The proposal in the paper was evaluated using the RAVDESS dataset. However, the paper does not provide information about the optimizer and learning rate.

Paper [26] uses the RAVDESS dataset to evaluate a model composed of a Convolutional Neural Network. The work compares feature extraction techniques and optimizers. In this context, they compare feature extraction techniques such as Mel frequency cepstral coefficients (MFCCs), Linear Prediction Cepstral Coefficients (LPCCs), the Wavelet Packet Transform (WPT), the Zero Crossing Rate (ZCR), Spectrum Centroid, Spectral Roll-off, Spectral Kurtosis, Root Mean Square (RMS), pitch, jitter, and shimmer. The optimizers used were SGD, Adam, and RMSProp, with a learning rate of 0.001.

Paper [27] compares three well-known 2D Convolutional Neural Network architectures in the literature: AlexNet, VGG16, and ResNet50. They use the RAVDESS dataset and a combination of RAVDESS+TESS+SAVEE+CREMAD to evaluate the model. They also use Mel spectrogram images without noise as a feature extraction technique. The authors do not specify which optimizer and learning rate were used, and data augmentation was not employed.

In [10], the combination of RAVDESS+TESS+SAVEE+CREMAD (R+T+S+C) datasets was used to assess how different feature extraction techniques influence the model's performance. They used MFCC, Mel, Chromagram, ZCR, RMS, and roll-off as the feature extraction techniques. They also investigated the influence of data augmentation by applying noise, stretching, and pitch. They compared a CNN, SVM, MLP, LSTM, and CNN+LSTM for the neural architecture. The optimizer used was Adam, with a variable learning rate.

Finally, in [28], the authors evaluate two neural architectures, a CNN and a CNN+LSTM. They also compare data augmentation with the addition of white noise and pitch. The feature extraction techniques used were ZCR, RMS, and MFCC combined. They used the RAVDESS, TESS, SAVEE, and CREMA-D datasets and their combination (R+T+S+C) to evaluate the methodology's performance. They used the SGD optimizer with a fixed learning rate of 0.001.

Based on the results described in Table 1, it is possible to observe some trends and gaps in some works in the literature in the SER research field. Of the eight works surveyed, six investigate the use of data augmentation, and most of the works compare the use and non-use of DA techniques, which are generally pitch and noise. Only the works of [23] and [25] carry out an investigation testing the operations separately and in combination. Another observation is that they attempt to evaluate and test different neural architectures for the SER problem. Of the eight works mentioned, six investigate the best architecture. The CNN algorithm and the combination of a CNN with LSTM or GRU are among the most used.

The first point we can see regarding the gaps in Table 1 is that the works in the SER area do not perform adjustments to the network's hyperparameters, such as analyzing the best optimizer and learning rate. Only [26] investigated the best optimizer, and no work analyzed the learning rate. The second aspect we can observe is the lack of work that evaluates and tests the best technique for extracting characteristics from audio for the SER problem. Among the eight works mentioned, only two papers ([10,26]) investigate the best feature extraction. The remaining papers use one or more techniques combined without evaluation. The third point is regarding the use of meta-learning. No work transfers configurations between databases for the problem of recognizing emotions in speech.

Therefore, this work's main contributions are to fill these gaps highlighted by the summary of works in Table 1. In this sense, we present the following original contributions:

1. A deep learning approach that considers the optimal combination of optimizer, learning rate, feature extraction, data augmentation, and neural architecture. No work in the literature includes an evaluation of all these aspects. It is also noteworthy that the authors developed the evaluated neural architectures of a CNN and a CNN+LSTM.
2. The use of meta-learning. This transfers settings for the optimizer, the learning rate, feature extraction, data augmentation, and the neural architecture between similar databases for the speech emotion recognition problem.

3. Deep Learning Approach for SER

Given the identified gaps in the literature, this study aims to present a deep learning approach aimed at identifying the most effective combinations for speech emotion recognition. To ensure replicability and explainability of the deep learning approach, a sequence of steps was developed that clearly outlines the decision-making process at each stage. As illustrated in Figure 1, the approach begins by selecting the databases to evaluate the method's performance. The RAVESS, TESS, SAVEE, and CREMA-D databases were chosen in this context.

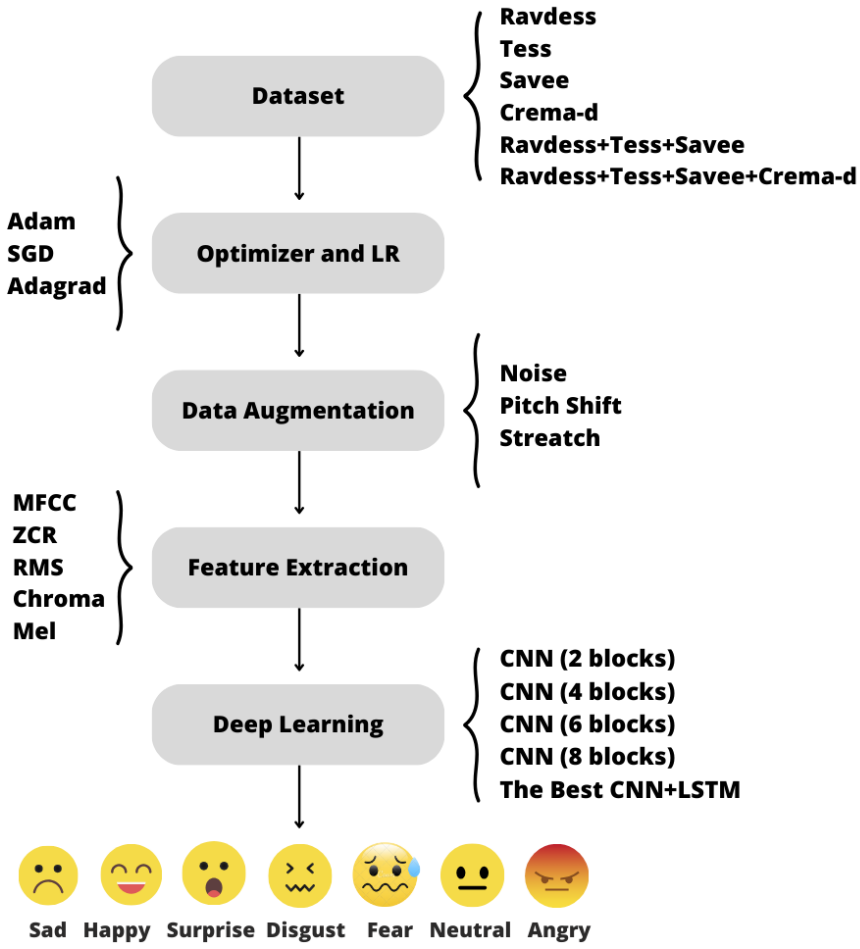

Figure 1. Flowchart of the proposed approach to seek the best speech emotion recognition configuration.

A standard framework was established for this research, utilizing a Convolutional Neural Network (CNN) with two blocks (Section 3.5 details its architecture). We employed the MFCC method for feature extraction and did not apply data augmentation. Based on this standard configuration, the first analysis focuses on the influence of the optimizer and the learning rate. It is crucial to emphasize that the learning rate significantly impacts learning performance, making it a critical hyperparameter in neural networks. Initially, we chose to concentrate our investigation on these two aspects. We evaluated three optimizers, Adam, SGD, and Adagrad, along with three distinct learning rates: 0.01, 0.001, and an adjustable rate.

After selecting the most suitable optimizer and the best learning rate for each dataset, the second phase of this research addresses the implementation of data augmentation. This technique is employed to increase the diversity of the audio samples, thereby enhancing the CNN's ability to generalize. In this context, various data augmentation techniques were evaluated, including noise addition, pitch shift, temporal stretching, and a combination of all these techniques together.

Next, each dataset was evaluated to identify the most effective audio feature extraction technique. This study involved testing the following feature extraction methods: Mel frequency cepstral coefficient (MFCC) values, the Zero Crossing Rate (ZCR), Root Mean Square (RMS), Chromagram, and Mel spectrogram.

Finally, we will evaluate deep learning architectures in the last stage. In this context, we proposed four configurations for the Convolutional Neural Network (CNN) with two blocks, four blocks, six blocks, and eight blocks. We will integrate the CNN architecture demonstrating the best performance into the LSTM algorithm. Thus, we will investigate the most effective CNN architecture and assess its performance when employing a hybrid approach (CNN+LSTM).

We will subject these configurations to a meta-learning evaluation after determining the best combinations of optimizer, learning rate, data augmentation, feature extraction methods, and deep learning architecture. In other words, we will transfer the combination that demonstrates the best overall performance for application to the CREMA-D and RAVDESS+TESS+SAVEE+CREMA-D datasets. This will enable us to apply the most effective approach to different datasets, allowing us to assess whether meta-learning is a suitable solution for speech emotion recognition (SER). The following sections will explain the steps of the proposed approach in more detail.

3.1. Dataset

The choice of a dataset plays a fundamental role in speech emotion recognition. The dataset selection directly impacts the model's training and generalization capabilities. Ideally, the dataset should encompass a broad spectrum of emotional expressions, various demographic factors, and diverse speaking styles to ensure the robustness and applicability of the model in real-world scenarios [22]. However, finding databases with a great diversity that are available online is not easy.

In this sense, this work uses four databases widely used in the literature, RAVDESS, TESS, SAVEE, and CREMA-D, to evaluate the performance of the proposed approach. Furthermore, to increase data availability for deep learning algorithms, the databases were also combined into two sets: R+T+S (RAVDESS+TESS+SAVEE) and R+T+S+C (RAVDESS+TESS +SAVEE+CREMAD). We will provide more detailed information about the databases used below.

RAVDESS: The Ryerson Audiovisual Database of Emotional Speech and Music (RAVDESS) [29] is a widely used resource in speech emotion recognition. It consists of recordings of 24 professional actors, divided equally between 12 women and 12 men. These actors make two statements each, both singing and speaking. Audios last 3 s and are labeled with emotions, happy, sad, angry, fearful, surprised, neutral, calm, and disgust, each presented in two levels of emotional intensity, normal and strong, totaling 2076 audio recordings. This work removed the calm emotion from the database to standardize it into seven emotions. Table 2 describes the audio distribution by emotion.

TESS: The Toronto Emotional Speech Set (TESS) [30] features recordings of two English actresses, one aged 26 and the other aged 64. The audios last two seconds; the labeled emotions are anger, disgust, fear, happiness, neutrality, surprise, and sadness. The dataset consists of 2800 audio files, with 400 audio recordings allocated to each emotion category, as illustrated in Table 2. It is worth mentioning that this database is balanced, guaranteeing an equal number of audio files for each emotion category.

SAVEE: The SAVEE (Surrey Audio–Visual Expressed Emotion) dataset [31] consists of 480 spoken audios by four English actors aged between 27 and 31. The audios last an average of 3 s and are labeled with seven emotions: anger, happiness, neutrality, disgust, sadness, fear, and surprise. However, it is important to note that this dataset presents a class imbalance problem. Specifically, the "neutral" class contains almost twice as many samples as all other classes combined, as illustrated in Table 2.

CREMA-D: The Crowdsourced Emotional Multimodal Actors (CREMA-D) dataset [32] encompasses 7442 unique audio samples, all recorded by 91 actors representing diverse racial and ethnic backgrounds. Among these actors, 48 were male and 43 were female; each uttered 12 sentences. The audios last 2 s on average and express six distinct emotions: anger, happiness, neutrality, disgust, sadness, and fear. In Table 2, it is possible to observe the number of audio recordings for each emotion within the database.

Table 2. Number of audios contained in each dataset.

Dataset	Happy	Sad	Angry	Fear	Disgust	Surprise	Neutral	Total
RAVDESS	376	376	376	376	192	192	188	2076
TESS	400	400	400	400	400	400	400	2800
SAVEE	60	60	60	60	60	60	120	480
CREMAD	1271	1271	1271	1271	1271	0	1087	7442
R+T+S	836	836	836	836	652	652	708	5356
R+T+S+C	2107	2107	2107	2107	1923	652	1795	12,798

3.2. Step 1: Tuning of Optimizers and the Learning Rate

Learning and optimization hyperparameters determine how the network learns and optimizes its parameters to achieve the minimum error. Among them are optimization algorithms and learning rates (LRs) [33]. Due to the importance of these hyperparameters for the CNN model, the first investigation into the proposed model seeks the best optimizer and learning rate for each database used. To this end, the three most used optimizers in the literature will be used, Adam, SGD, and Adagrad, with the learning rates most found in articles in the SER research field (0.01, 0.001, and adaptive learning rate with a factor of 0.4 and a minimum LR of 0.000001) [10,15,28]. Below is a brief explanation of the optimizers and learning rates used.

Stochastic Gradient Descent (SGD) is the optimization technique most widely employed in machine learning, especially deep learning. Unlike regular gradient descent, which computes the loss and gradient over the entire training dataset before adjusting parameters, SGD adopts a more nimble approach. It randomly selects one data point from the training set for each step and computes the gradient using only that instance [33].

The Adam optimizer, short for Adaptive Moment Estimation, is a widely used optimization algorithm in machine learning and neural network training. Adam combines concepts from Stochastic Gradient Descent (SGD) and the momentum method by computing moving averages of past and squared gradients. This enables it to handle non-stationary optimization problems effectively and helps prevent undesired oscillations in the convergence path. Adam has proven to be particularly efficient in accelerating the training of deep networks and is widely embraced in the machine learning community [33].

Adagrad, an abbreviation for "Adaptive Gradient Descent," is an optimization technique based on gradient descent. It is an optimizer that utilizes learning rates tailored to specific parameters, adapting them according to how often each parameter is updated during training. Parameters subject to frequent updates experience reduced learning rates, leading to progressively smaller parameter adjustments as the training unfolds [34].

One of the optimizer's input parameters is the learning rate (LR). Theoretically, a minimal learning rate guarantees the minimum error (if training for an infinite time). A high learning rate speeds up learning but does not guarantee finding the minimum error [33].

3.3. Step 2: Data Augmentation Optimization

The second phase of the proposed deep learning approach involves investigating the impact of data augmentation on the performance of the deep learning algorithm. Data augmentation generates new synthetic training samples through slight perturbations of existing examples. We aim to make our model invariant to these perturbations and enhance its generalization capacity. Various techniques are used to augment audio data, with some of

the most common ones including noise, time stretching, and pitch variation [10]. The visual representation of these techniques' effects is illustrated in Figures 2–5.

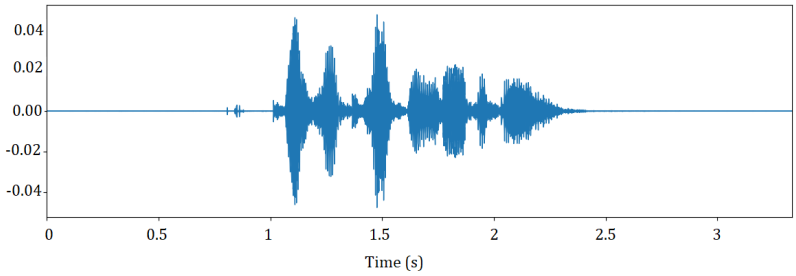

Figure 2. The original sound waveform.

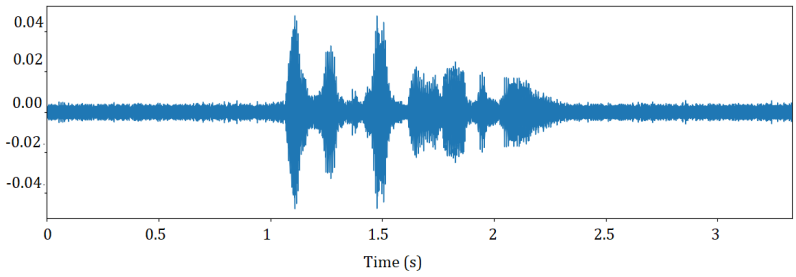

Figure 3. Noise.

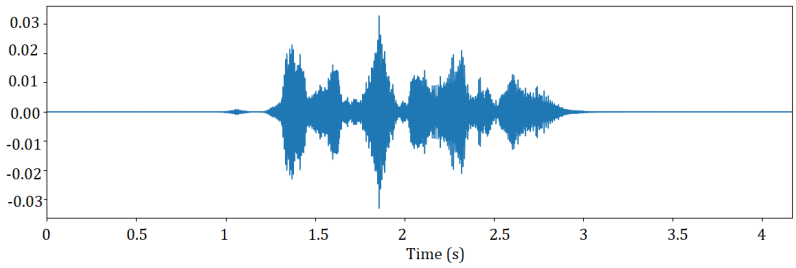

Figure 4. Stretch.

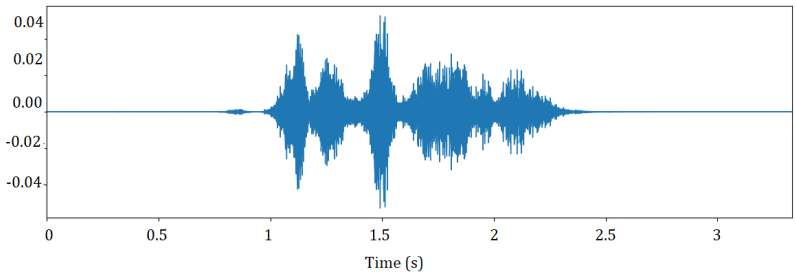

Figure 5. Pitch.

In this study, the noise injection technique was employed to introduce random values into the data using both NumPy's normal and uniform methods with a rate of 0.035 [35]. The time-stretching technique was also used to elongate time series at a fixed rate of 0.8, implemented through the time-stretching method from the Python library Librosa [22]. Lastly, random pitch alterations were applied with a pitch-shifting factor of 0.7, using Librosa's pitch-shifting method [22]. We tested the data augmentation techniques both individually and in combination.

3.4. Step 3: Feature Extraction Seletion

The extraction of features from speech audio signals constitutes a fundamental step in speech emotion recognition (SER) activities [36]. The third step of this investigation specifically employs the five most used spectral attributes in the SER research field [10,22,24,28]. They are Mel frequency cepstral coefficient (MFCC) values, the Zero Crossing Rate (ZCR), Chromagram, Mel spectrogram, and Root Mean Square (RMS) values. More details of each feature extraction technique will be presented below.

3.4.1. Mel Frequency Cepstral Coefficients (MFCC)

To derive MFCC features, the initial step involves dividing the speech signal into short frames of 20–30 ms each, advanced every 10 ms to capture temporal features of individual speech signals. The Discrete Fourier Transform (DFT) is subsequently applied to each windowed frame, converting them into magnitude spectra. Next, 26 filters are employed on the signal obtained in the previous step to compute the Mel-scaled filter bank (MSFB). The MSFB, grounded in human ear frequency perception, yields 26 values describing the energy of each frame. Log energies are computed to obtain log filter bank energies. Equation (1) quantifies the Mel estimation from a physical frequency [10,22,37]:

$$f_{Mel} = 2590 \log_{10} 1 + \frac{f}{700} \tag{1}$$

Here, f denotes the physical frequency (in Hz) and f_{Mel} represents the frequency perceived by the human ear. After obtaining log filter bank energies, the Discrete Cosine Transform (DCT) is applied to generate the MFCCs [12,22]. The extraction of MFCC values from the datasets was performed using the Librosa library.

3.4.2. Zero Crossing Rate (ZCR)

The ZCR is a commonly used feature in SER. It quantifies the number of times the amplitude of a speech signal crosses the zero-value threshold within a specified time frame. The ZCR has been proven effective in distinguishing between voiced and unvoiced expressions. Mathematically, the ZCR is defined by Equation (2), where s represents a signal of length T, and $1_{\mathbb{R}<0}$ is an indicator function. ZCR values from the datasets were extracted using the Librosa library [22].

$$ZCR = \frac{1}{T-1} \sum_{t=1}^{T-1} 1_{\mathbb{R}<0}(S_t S_{t-1}) \tag{2}$$

3.4.3. Chromagram

The Chromagram (Chroma) feature characterizes the tonal content of an audio signal, closely related to the 12 classes of pitch. Chroma features excel at capturing harmonic and melodic audio traits. Chromagram features are derived by applying Short-Time Fourier Transforms (STFTs) to the audio waveform from the dataset [38]. The extraction of Chroma values from the datasets was carried out using the Librosa library.

3.4.4. Mel Spectrogram

A spectrogram visualizes a signal's frequency spectrum over time through Fast Fourier Transform (FFT) analysis. It divides the frequency spectrum into Mel scale frequencies, producing a Mel spectrogram for each window. Magnitude components corresponding to the Mel frequencies are then isolated [10,38]. In this study, these values were extracted from the datasets using the Librosa library.

3.4.5. Root Mean Square (RMS) Value

The RMS value is computed for each frame of speech audio samples, offering an average signal amplitude irrespective of positive or negative amplitude levels. For a given

signal $x = x_1, x_2, x_3, \ldots, x_n$, the RMS value x_{RMS} can be determined using Equation (3) [22]. RMS values were extracted from the datasets using the Librosa library.

$$x_{RMS} = \sqrt{\frac{x^2}{n}} = \sqrt{\frac{1}{n}(x_1^2, x_2^2, x_3^2, \ldots, x_n^2)} \tag{3}$$

3.5. Step 4: Neural Architecture Search

The fourth stage of the proposed methodology involves investigating the best architecture. In this regard, two well-known algorithms in the literature were utilized: the Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). Below, we will provide more information on these two deep learning algorithms and how they will be used in this study.

3.5.1. Convolutional Neural Network

In this study, we will employ Convolutional Neural Networks (CNNs) to classify emotions based on speech data. We will identify the most effective architectures for each dataset to achieve this goal. The proposed architectures are illustrated in Figure 6. Initially, we will conduct tests using the two-block CNN architecture. In the final phase of our methodology, we will investigate the optimal number of blocks for model performance, considering two blocks, four blocks, six blocks, and eight blocks.

Each CNN block includes a convolutional layer (1D) with ReLu activation, batch normalization, max pooling 1D (pool size = 5), and dropout (rate = 0.2). Batch normalization is a layer that normalizes the inputs by applying a transformation that keeps the average output close to 0 and the standard deviation of the output close to 1. Max pooling is a technique used to reduce the spatial dimensionality of the feature representation and to maintain the most relevant characteristics. Furthermore, dropout is a regularization technique to avoid overfitting in neural networks.

At the end of the CNN blocks, a flatten layer and two dense layers were added to perform the final classification. The first dense layer contains the ReLu activation function, and the output dense layer contains the softmax activation function. The Rectified Linear Unit (ReLU) activation function activates a node only if the input is above zero. If the input is below zero, the output is always zero. However, when the input exceeds zero, it has a linear relationship with the output variable. The ReLU function is represented by [33]:

$$f(x) = max(0, x). \tag{4}$$

The softmax function is a generalization of the sigmoid function. It obtains classification probabilities when there are more than two classes. It transforms input values into probability values between 0 and 1, where the final sum of all probabilities is 1. A prevalent use case in deep learning problems is to predict a single class among many options (more than two) [33]. The softmax equation is described by (5) [33], where $\sigma(x_j)$ is the probability of the output neuron, x_j is the output neuron's vector, and i are the indices of all neurons.

$$\sigma(x_j) = \frac{e^{x_j}}{\sum_i e^{x_i}} \tag{5}$$

In this work, fixed hyperparameters were used, including 100 epochs and a batch size of 64. For the loss function, we employed "categorical crossentropy," which can be represented by Equation (6) [33], where $L(y, \hat{y})$ is the value of the loss function, y_i represents the actual probability of class i (a binary value, 0 or 1, indicating the correct class), and $\hat{y}_i$ represents the predicted probability for class i by the model.

$$L(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i) \tag{6}$$

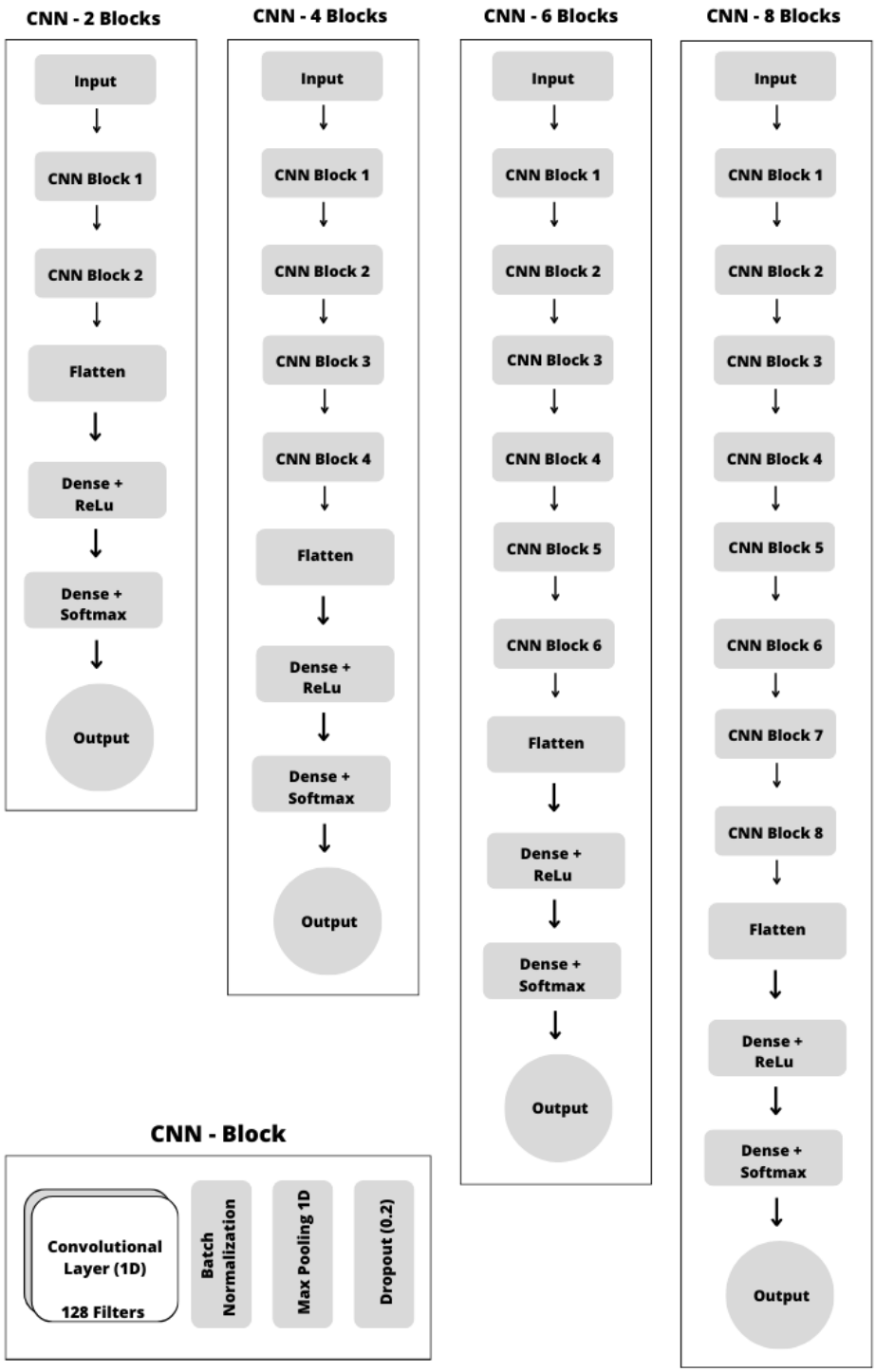

Figure 6. Convolutional Neural Network architectures for comparison.

3.5.2. Long Short Term Memory

Long Short-Term Memory (LSTM) is a Recurrent Neural Network (RNN) architecture that handles sequential data and is widely used in natural language processing, voice recognition, and time series prediction tasks [39]. After selecting the best CNN architecture, we will investigate the model performance by adding two LSTM layers. Figure 7 shows that the first layer has 258 units and the second LSTM layer has 128 units.

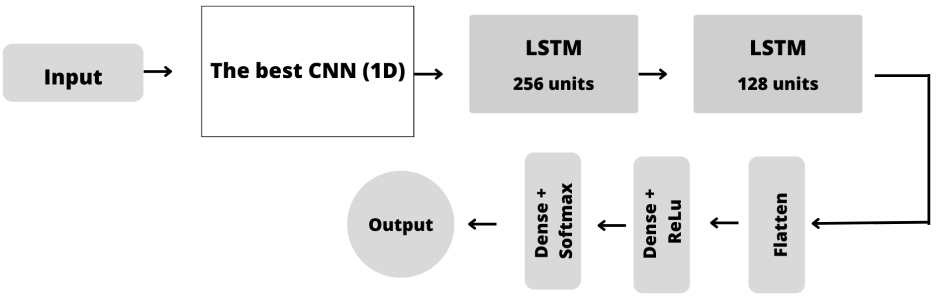

Figure 7. Hybrid architecture with a Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM).

3.6. Meta-Learning

Meta-learning (MtL) is a process of using knowledge acquired from a specific dataset and transferring it to new databases [19]. The transferred knowledge, also known as meta-knowledge, can include information such as neural architectures, resulting models, and configurations for previously obtaining better models [40]. In this way, it is possible to explore how to learn from past experiences and reduce the computational time and cost required for model adaptation [17].

Figure 8 illustrates how meta-learning was utilized in this work. As described in the previous sections, the RAVDESS, TESS, SAVEE, and R+T+S databases were initially selected to adjust the optimizer and learning rate settings, data augmentation, feature extraction, and neural architecture. After finding the best configuration (i.e., the set of configurations that has the best accuracy value for most of the datasets), it is then stored in a knowledge base.

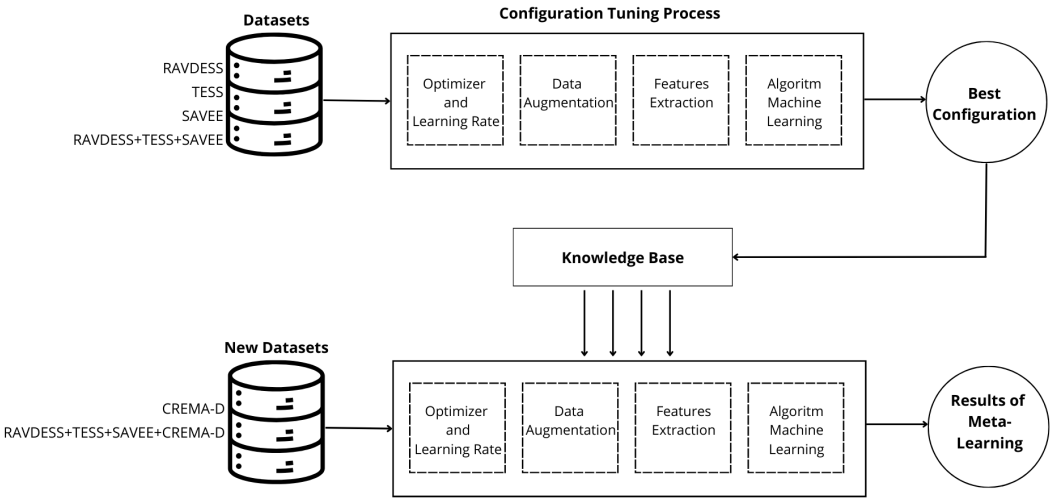

Figure 8. Meta-learning system to transfer SER configurations (optimizer, learning rate, data augmentation, feature extraction, and neural architecture) between different databases.

After the best configuration is found and stored in the knowledge base, meta-learning is carried out, in which new datasets, CREMA-D and R+T+S+C (RAVDESS+TESS+SAVEE+CREMA-D), will access the best configurations of the knowledge base and apply the same choices of the optimizer, learning rate, data augmentation, feature extraction, and neural architecture in the deep learning process. The result obtained through meta-learning is then compared with not using meta-learning, that is, carrying out the entire approach proposed for the CREMA-D and R+T+S+C bases.

4. Results

This section will present the results obtained using the proposed methodology described in Section 3. To conduct the experiments, we used a notebook with a Windows

11 operating system, an Intel i5 1135G7 2.40 GHz processor, 8 GB of RAM, and an Nvidia GeForce MX350 GPU with 2 GB VRAM. The development environment used in this work consists of a Jupyter IDE, associated with the Python language.

As described in Section 3, the datasets used to evaluate the methodology's performance were RAVDESS, TESS, SAVEE, and RAVDESS+TESS+SAVEE (R+T+S). Subsequently, the combinations showing the best accuracy performance will be transferred to the CREMA-D and RAVDESS+TESS+SAVEE+CREMA-D (R+T+S+C) datasets. All the datasets used were divided into 80% for training, 10% for validation, and 10% for testing.

For the first step, we adjusted each dataset's optimizer and learning rate. In this regard, we used the Adam, SGD, and Adagrad optimizers, combined with learning rates of 0.01, 0.001, and an adjustable rate. The results for this initial investigation can be seen in Table 3, which presents the test accuracy values for each evaluated dataset.

In Table 3, one can observe how the variation in hyperparameters influences the classifier's performance. When analyzing the RAVDESS database, it was observed that the variation in optimizer and learning rate leads to a wide improvement in precision values, ranging from 30.76% (with Adam 0.01) to 80.01% (with variable Adam). This influence extends to other databases.

Table 3. Results of accuracy (%) obtained by adjusting the optimizer and learning rate. The most accurate values for each dataset are emphasized in bold. RAVD is RAVDESS and R+T+S is RAVDESS+TESS+SAVEE.

Dataset	Adam			SGD			Adagrad		
	0.01	0.001	Variable	0.01	0.001	Variable	0.01	0.001	Variable
RAVD.	30.76	78.84	**80.01**	75.07	52.88	60.09	77.88	50.00	42.78
TESS	98.21	98.57	**100.00**	99.64	98.93	98.02	99.64	99.29	99.29
SAVEE	64.58	**70.83**	68.75	66.67	43.75	62.50	58.33	54.17	56.25
R+T+S	67.54	**87.31**	87.13	86.57	77.05	84.51	86.94	76.49	74.63

For the next steps, the best combinations of optimizer and learning rate were used for each dataset. Consequently, the RAVDESS and TESS datasets employed variable Adam, while the SAVEE and R+T+S datasets used Adam with a learning rate of 0.001.

The second stage of the methodology involves the use of data augmentation. In this regard, three techniques were investigated: noise, stretch, and pitch, as well as the combination of all three. Table 4 presents the accuracies from the previous investigation, i.e., without data augmentation (no D.A). It also displays the accuracies for each technique investigated. As seen in Table 4, the stretch technique performed well in most datasets. Only the Tess dataset maintained 100% accuracy across all combinations in this stage.

Table 4 reveals the impact of data augmentation, highlighting that for the RAVDESS dataset, the precision varies from 77.88% (using only noise) to 96.63% (using only stretch). In contrast, the SAVEE dataset ranges from 67.50% (noise) to 85.83%, indicating an enhancement in accuracy. Notably, the R+T+S dataset shows an improvement of around 10% when applying the stretch operation.

Table 4. Accuracy (%) achieved for different data augmentation techniques, noise, pitch variation, and stretch, across four distinct datasets. The highest accuracy values for each dataset are emphasized in bold.

Dataset	no D.A	Noise	Streatch	Pitch	All D.A.
RAVDESS	80.01	77.88	**96.63**	86.30	85.23
TESS	100.00	100.00	100.00	100.00	100.00
SAVEE	70.83	67.50	**85.83**	80.00	81.94
R+T+S	87.31	88.99	**96.64**	93.47	92.21

For the third stage, stretch-type data augmentation was applied to all datasets to investigate the best audio feature extraction technique. The standard technique used in this article is MFCCs, and in this stage, other techniques such as Chroma, ZCR, RMS, Mel, and the application of all together will be explored. Table 5 shows the test accuracy values obtained in this third stage. As can be observed, the MFCC technique performed better in most datasets, except for the SAVEE dataset, which achieved a higher accuracy with all the techniques applied together. Furthermore, it is possible to observe that the selection of the feature extraction method significantly impacts the SER classification. Table 5 reveals, for example, that in the RAVDESS database, the accuracy is 20.91% when using the ZCR and increases to 96.63% when using MFCCs. Similarly, other databases show significant variations in accuracy depending on the resource extraction technique used.

Table 5. Accuracy (%) obtained for different feature extraction techniques, including MFCC, Chroma, ZCR, RMS, and Mel, across four distinct datasets. Bold accuracies represent the highest values.

Dataset	MFCC	Chroma	ZCR	RMS	Mel	All
RAVDESS	**96.63**	63.70	20.91	30.70	67.79	94.23
TESS	**100.00**	89.82	19.82	32.14	98.57	99.08
SAVEE	85.83	52.50	32.08	32.50	70.42	**90.62**
R+T+S	**96.64**	73.41	19.96	24.63	83.68	96.36

The fourth stage of the methodology, the neural architecture search, aims to investigate the optimal architecture for the Convolutional Neural Network proposed in this paper. In this regard, four architectures were tested, two blocks, four blocks, six blocks, and eight blocks, as described in Section 3.5.

In Table 6, the test accuracy values for each architecture can be observed. The RAVDESS and R+T+S datasets achieved better accuracies with the four-block CNN, reaching 96.88% and 97.11%, respectively. The SAVEE dataset achieved an accuracy of 90.62% with the two-block CNN. However, the TESS dataset did not show any variation in the accuracy values at this stage of the methodology.

Table 6. Accuracy (%) obtained for different CNN architectures, two blocks, four blocks, six blocks and eight blocks, on four different datasets. Accuracies in bold represent the highest values.

Dataset	CNN (Two Blocks)	CNN (Four Blocks)	CNN (Six Blocks)	CNN (Eight Blocks)
RAVDESS	96.63	**96.88**	95.29	95.19
TESS	100.00	100.00	100.00	100.00
SAVEE	**90.62**	87.50	76.04	83.33
R+T+S	96.64	**97.11**	96.92	95.18

After determining the best CNN architecture for each dataset, i.e., the most effective CNN architecture identified in the previous stage, two LSTM layers were added to assess whether a hybrid deep learning algorithm improves the test accuracy. Table 7 showcases the test accuracy results for both the optimal CNN architecture and the hybrid CNN+LSTM architecture, organized by dataset. In the case of the RAVDESS and R+T+S datasets, the hybrid architecture (CNN (four blocks) + LSTM) consistently exhibited a superior accuracy. However, for the SAVEE dataset, the two-block CNN architecture yielded better results. Remarkably, the Tess dataset showed no significant changes.

Table 7. Accuracy (%) obtained for the best CNN architecture and hybrid architecture (CNN+LSTM) on four distinct datasets. Accuracies in bold represent the highest values.

Dataset	Best CNN	Best CNN+LSTM
RAVDESS	96.88	**97.01**
TESS	100.00	100.00
SAVEE	**90.62**	85.42
R+T+S	97.11	**97.37**

Upon completing the methodology steps, it became possible to describe the best combination for each dataset, considering the optimizer, learning rate, data augmentation techniques, feature extraction methods, and neural architecture that yielded the highest performance. For the RAVDESS dataset, the best combination involved using the Adam optimizer with a variable learning rate, applying stretch for data augmentation, MFCCs for audio feature extraction, and a hybrid architecture consisting of a four-block CNN combined with LSTM. Regarding the computational time spent, the code was executed 25 times for each database, totaling 150 executions due to working with six databases. The duration of each execution varies mainly according to the database (number of audio files) and the neural architecture used. Therefore, it can be stated that there was an average computational cost of 25 h incurred.

In Figure 9, we can observe the graphs illustrating the accuracy and loss trends throughout the training and validation processes of the RAVDESS dataset. Additionally, Table 8 displays the precision, recall, and F-score metrics derived from the optimized configuration applied to the RAVDESS dataset.

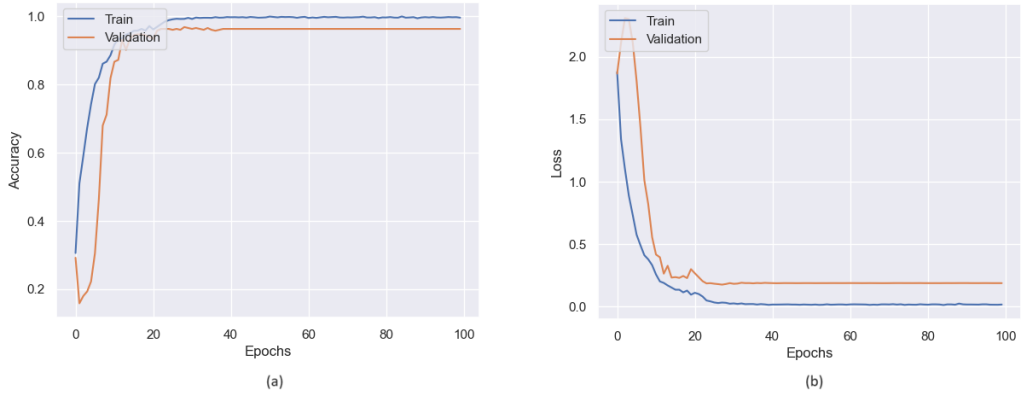

Figure 9. Graph of (**a**) accuracy and (**b**) loss values during training and validation of the RAVDESS dataset.

Table 8. Precision, recall, and F-score percentage values for the RAVDESS database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	95.0	93.0	94.0
Happy	95.0	99.0	97.0
Fear	100.0	93.0	96.0
Neutral	100.0	100.0	100.0
Angry	94.0	99.0	96.0
Surprise	97.0	97.0	97.0
Sad	98.0	98.0	98.0
Accuracy	-	-	97.0

The TESS dataset achieved a 100% accuracy rate using a combination of the Adam optimizer with a variable learning rate, MFCC feature extraction, no data augmentation, and a two-block CNN architecture. In several stages, the accuracy remained unchanged,

leading us to consider the initial combination, which achieved 100% accuracy as the best combination. In Figure 10, we can observe the training and validation accuracy and loss curves for the TESS dataset. In Table 9, we can find the precision, recall, and F-score metrics calculated using the optimized combination for the TESS dataset.

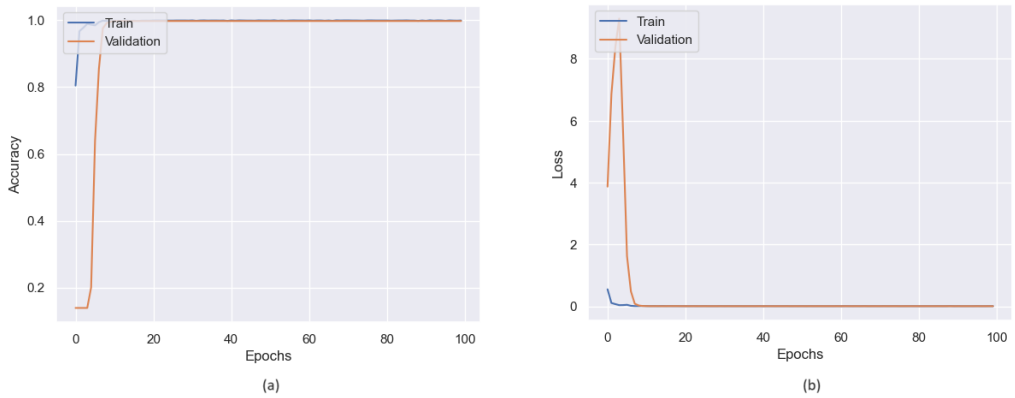

Figure 10. Graph of (**a**) accuracy and (**b**) loss values during training and validation of the TESS dataset.

Table 9. Precision, recall, and F-score percentage values for the TESS database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	100.0	100.0	100.0
Happy	100.0	100.0	100.0
Fear	100.0	100.0	100.0
Neutral	100.0	100.0	100.0
Angry	100.0	100.0	100.0
Surprise	100.0	100.0	100.0
Sad	100.0	100.0	100.0
Accuracy	-	-	100.0

The SAVEE dataset achieved a maximum accuracy of 90.62% with a combination of the Adam optimizer, a learning rate of 0.001, stretch-type data augmentation, feature extraction using all techniques together (MFCC, ZCR, RMS, Chroma, and Mel), and a neural architecture consisting of a two-block CNN. Figure 11 displays the training and validation accuracy and loss history. Additionally, Table 10 presents the precision, recall, and F-score values obtained.

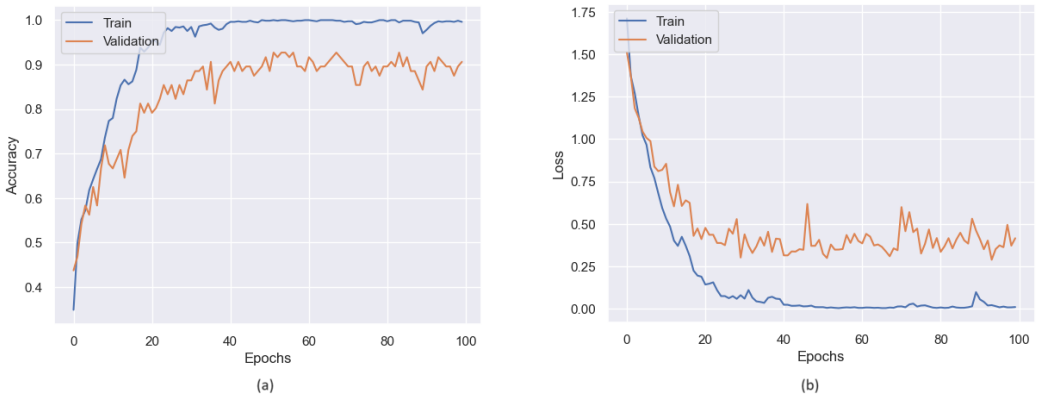

Figure 11. Graph of (**a**) accuracy and (**b**) loss values during training and validation of SAVEE dataset.

Table 10. Precision, recall, and F-score percentage values for the SAVEE database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	100.0	91.0	95.0
Happy	86.0	86.0	86.0
Fear	92.0	100.0	96.0
Neutral	100.0	86.0	92.0
Angry	92.0	92.0	92.0
Surprise	93.0	87.0	90.0
Sad	73.0	100.0	85.0
Accuracy	-	-	90.6

The dataset that combines RAVDESS+TESS+SAVEE achieved a maximum accuracy of 97.37% with a combination of the Adam optimizer, a learning rate of 0.001, stretch-type data augmentation, MFCC feature extraction, and a neural architecture consisting of a hybrid CNN with four blocks and LSTM. Figure 12 presents the training and validation accuracy and loss history, and Table 11 displays the precision, recall, and F-score values obtained.

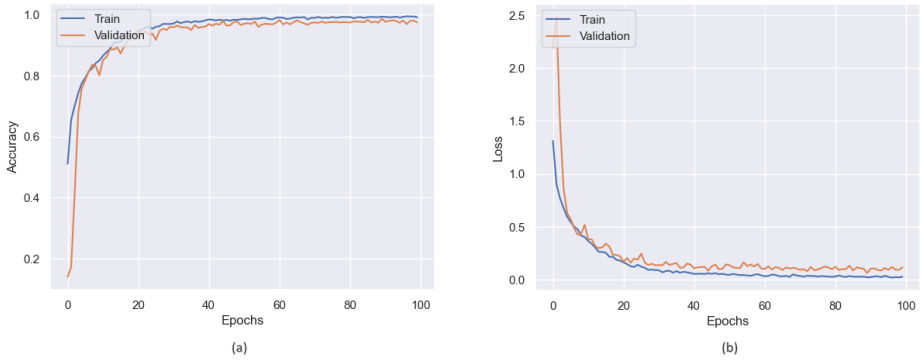

Figure 12. Graph of (**a**) accuracy and (**b**) loss values during training and validation of the RAVDESS+TESS+SAVEE dataset.

Table 11. Precision, recall, and F-score percentage values for the RAVDESS+TESS+SAVEE database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	98.0	96.0	97.0
Happy	96.0	96.0	96.0
Fear	95.0	95.0	95.0
Neutral	95.0	100.0	97.0
Angry	97.0	95.0	96.0
Surprise	96.0	99.0	97.0
Sad	98.0	93.0	95.0
Accuracy	-	-	97.4

4.1. Results of Meta-Learning

This section provides the results of the meta-learning carried out. As illustrated in Figure 8, once the optimizer, learning rate, data augmentation, feature extraction, and neural architecture settings were identified for the RAVDESS, TESS, SAVEE, and R+T+S databases, the best configuration found, that is, the configuration that obtains the highest accuracy values for the majority of datasets, was subsequently registered in a knowledge base. In this way, a comparison is made between using the configurations contained in the knowledge base, transferring the configurations to the new CREMA-D and R+T+S+C databases, that is, using meta-learning ("yes meta-learning") with the non-use of meta-learning ("no meta-learning"), in which the CREMA-D and R+T+S+C databases go through the entire step-by-step approach proposed in Section 3. The comparison will be based on the accuracy value and the computational time.

In Table 12, a comparison between the "no meta-learning" configurations and those derived from the meta-learning process, "yes meta-learning," is provided, along with the corresponding accuracy values and computational time consumed. In this context, we can observe that the configurations are quite similar. For the CREMA-D dataset, the difference between the "no meta-learning" and "yes meta-learning" configurations pertained only to the optimizer and learning rate, increasing accuracy by 2.16%. Additionally, meta-learning significantly reduced the computational time required from 218:51 to 15:19 min.

Table 12. Comparison of accuracy values (%) and computational time (min) for the CREMA-D and R+T+S+C databases using the "no meta-learning" and "yes meta-learning" configurations. Bold values indicate the best result in accuracy and computational time.

Datasets	Meta-Learning	Optimizer	L.R.	D.A.	Features	Arch.	Acc. (%)	Time (min)
CREMAD	No	Adagrad	0.01	Stretch	MFCC	CNN4b+ LSTM	81.12	218:51
	Yes	Adam	variable	Stretch	MFCC	CNN4b+ LSTM	**83.28**	**15:19**
R+T+S+C	No	Adagrad	0.01	Stretch	MFCC	CNN6b	86.87	388:32
	Yes	Adam	variable	Stretch	MFCC	CNN4b+ LSTM	**90.94**	**25:50**

For the R+T+S+C dataset, the difference between the "no meta-learning" and "yes meta-learning" configurations involved changes in the optimizer, learning rate, and neural architecture. This difference increased the accuracy by 4.07%. Furthermore, meta-learning significantly reduced the computational time required, as observed, from 388:32 min to 25:50 min.

In Figure 13, one can examine the training and validation history of (a) accuracy and (b) loss for the CREMA-D dataset. It is evident that over the course of 100 epochs, the model successfully generalized emotions from the dataset. In Table 13, one can verify the precision, recall, and F-score data obtained through the meta-learning process.

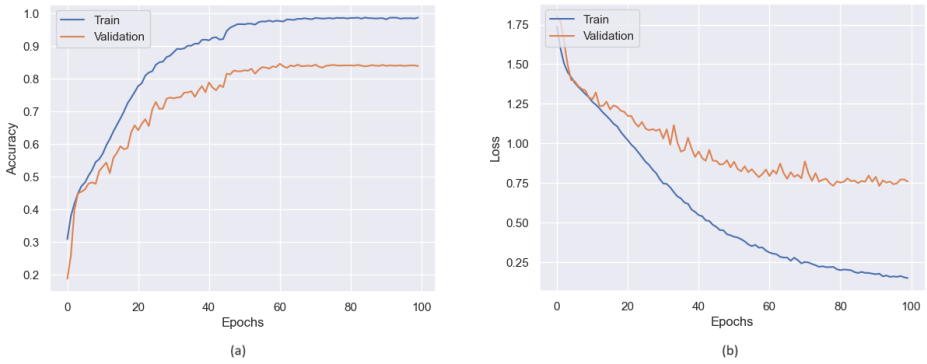

Figure 13. Graph of (**a**) accuracy and (**b**) loss values during training and validation of the CREMA-D dataset.

In Figure 14, one can observe the training and validation history of (a) accuracy and (b) loss for the RAVDESS+TESS+SAVEE+CREMA-D datasets, allowing for an observation of the model's generalization across all seven classes within the dataset. In Table 14, the data for precision, recall, and F-score obtained for the CREMA-D dataset can be examined.

Table 13. Precision, recall, and F-score percentage values for the CREMA-D database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	81.0	84.0	82.0
Happy	80.0	83.0	82.0
Fear	85.0	80.0	82.0
Neutral	76.0	80.0	78.0
Angry	91.0	87.0	89.0
Sad	86.0	85.0	85.0
Accuracy	-	-	83.3

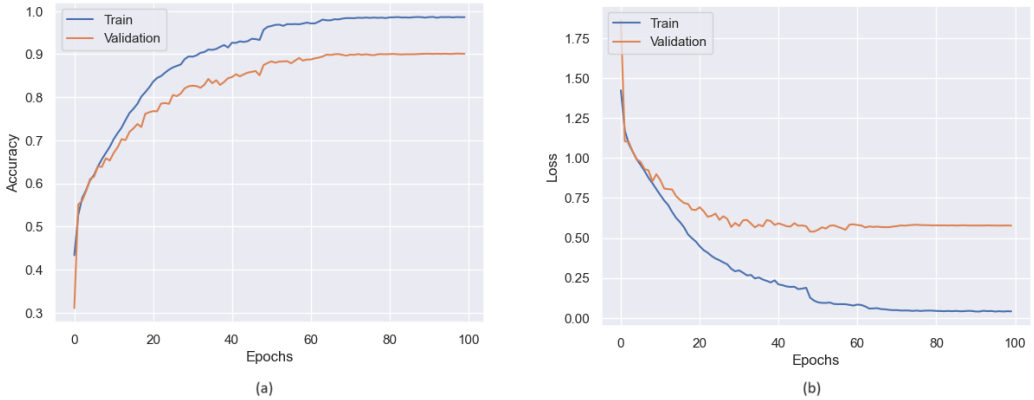

Figure 14. Graph of (**a**) accuracy and (**b**) loss values during training and validation of the RAVDESS+TESS+SAVEE+CREMA-D dataset.

Table 14. Precision, recall, and F-score percentage values for the RAVDESS+TESS+SAVEE+CREMA-D database.

Emotions	Precision (%)	Recall (%)	F-Score (%)
Disgust	95.0	93.0	94.0
Happy	95.0	99.0	97.0
Fear	100.0	93.0	96.0
Neutral	100.0	100.0	100.0
Angry	94.0	99.0	96.0
Surprise	97.0	97.0	97.0
Sad	98.0	98.0	98.0
Accuracy	-	-	97.0

4.2. Comparison of Results

This section aims to conduct a comparative analysis of the results obtained in relation to prior research documented in the literature. We have referred to the studies outlined in Section 2 to accomplish this. Table 15 provides a comprehensive overview of the studies under examination, the datasets employed, and the corresponding accuracy values reported in each article.

Most of the papers mentioned in Table 15 use the RAVDESS database. When comparing the accuracy values obtained, it is evident that our method achieves a higher accuracy, at 97.01%. The works in [22,23] come closest to our result, with 95.52% and 95.22% accuracies, respectively. The same is true when we analyze the TESS database, where our work achieved 100% accuracy.

The SAVEE and CREMA-D databases performed better in the work in [22], which uses a hybrid architecture with CNN, LSTM, and GRU as the classifier. The model in [22] achieved an accuracy of 93.22% for SAVEE and 90.47% for CREMA-D. Our work obtained a similar accuracy for SAVEE, at 90.47%, while our work achieved 83.28% for the CREMA-D database.

Table 15. Comparison of accuracy values (%) of the proposed approach with related work. Bold values indicate the best result of accuracy.

Paper	RAVDESS	TESS	SAVEE	CREMA-D	R+T+S	R+T+S+C
[23]	95.52	-	-	-	-	-
[22]	95.22	99.46	**93.22**	**90.47**	-	-
[24]	68.00	-	-	-	89.00	-
[25]	89.33	-	-	-	-	-
[26]	94.18	-	-	-	-	-
[27]	77.31	-	-	-	89.93	72.94
[10]	-	-	-	-	-	92.73
[28]	92.60	99.60	84.90	89.90	-	**94.50**
Proposed	**97.01**	**100.00**	90.62	83.28	**97.37**	90.94

Regarding the databases that combined other datasets, such as RAVDESS+TESS+SAVEE (R+T+S) and RAVDESS+TESS+SAVEE+CREMA-D (R+T+S+C), our work achieved an accuracy of 97.37% for the R+T+S dataset, standing out compared to the mentioned works. In the R+T+S+C database, paper [28] achieved an accuracy of 94.50% using a hybrid network composed of a CNN and LSTM.

This paper stands out for carrying out experiments on six databases, with the aim of evaluating the robustness and consistency of the proposed approach. Although all databases are related to emotion recognition in speech, each dataset presents distinct characteristics and challenges. In this aspect, it is noteworthy that the proposed approach demonstrated superior performance for three analyzed datasets (RAVDESS, TESS and R+T+S), that is, for half of the problems applied. Thus, we achieved a superior performance increase of 1.49% in accuracy for RAVDESS, an increase of 0.40% for TESS, and an increase of 7.44% in accuracy for the R+T+S dataset.

Furthermore, the proposed approach achieved accuracies greater than 90% in five of the analyzed datasets. For example, compared to the work in [27], experiments were carried out with three datasets, but none of them achieved an accuracy greater than 90%. Likewise, paper [24] evaluated two databases and neither of them achieved an accuracy greater than 90%. In this way, the results in Table 15 reinforce the robustness of the proposed approach, given the consistency of the accuracy results for the evaluated speech emotion recognition problems.

5. Conclusions

This work aimed to develop a deep learning approach for assessing the most effective configurations for speech emotion recognition. Additionally, we sought to conduct a meta-learning analysis related to optimizers, learning rates, data augmentation techniques, feature extraction strategies, and neural architectures. The transfer of configurations was performed across different SER datasets.

The proposed approach produced results that are close to those in the literature. Accuracy results in the datasets were impressive: RAVDESS achieved 97.01%, TESS 100%, SAVEE 90.62%, and R+T+S 97.37%. The best overall configuration involved using the Adam optimizer with a variable learning rate, applying data augmentation in the stretch format, feature extraction in the MFCC style, and implementing a hybrid neural architecture composed of a CNN4b and LSTM.

After identifying the optimal configuration, these settings were transferred to the knowledge base and applied to the new datasets, CREMA-D and R+T+S+C. Subsequently, the meta-learning process was conducted. The results demonstrate that meta-learning outperformed methodical results. CREMA-D showed a 2.16% increase in accuracy, while R+T+S+C achieved a 4.07% improvement. This suggests the feasibility of applying meta-learning to various speech emotion recognition (SER) datasets, considering the optimizer, the learning rate, data augmentation, feature extraction, and the neural architecture.

This work has aspects that can be worked on in the future for improvement. One of these topics is including and contributing to large language models (such as ChatGPT). One of the relevant issues is addressing cultural differences in the recognition of speech emotions, and the only dataset used in this paper that has cultural diversity is CREMA-D. Another relevant issue to be worked on is improving the proposed approach by including new parameters to be analyzed, such as the number of epochs, batch size, application of cross-validation, and others. Finally, emotion recognition is a broad area, and multimodal methods can be inserted for emotional recognition with greater assertiveness, i.e., using speech, facial expressions, electroencephalogram signals, and other modals.

Author Contributions: L.T.C.O.: Methodology, Experiments, Writing—Original Version. A.L.C.O.: Methodology, Writing—Original Version. J.d.J.F.C.: Supervision, Writing—Review and Editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by FAPESB (084.0508.2022.0000201-09), CAPES (Financing Code 001), UFBA and UFRB.

Institutional Review Board Statement: Ethical review and approval were waived for this study, because the datasets used are accessible under request for research purposes and the authors of this work adhered to the terms of the license agreement of the first dataset.

Informed Consent Statement: Subject consent was waived because the datasets used are accessible under request for research purposes and the authors of this study adhered to the terms of the license agreement of the datasets.

Data Availability Statement: The RAVDESS database used in this paper is available upon request from https://zenodo.org/record/1188976#.YTscC_wzY5k, accessed on 25 November 2022.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SER	Speech Emotion Recognition
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
DNN	Deep Neural Network
HCI	Human–Computer Interaction
DA	Data Augmentation
LR	Learning Rate
MtL	Meta-Learning
ML	Machine Learning
MFCC	Mel Frequency Cepstral Coefficient
ZCR	Zero Crossing Rate
RMS	Root Mean Square

References

1. Ottoni , L.T.C.; Cerqueira, J.J.F. A Review of Emotions in Human-Robot Interaction. In Proceedings of the 2021 Latin American Robotics Symposium (LARS), Natal, Brazil, 11–15 October 2021; pp. 7–12.
2. Oliveira, M.L.L.; Cerqueira, J.J.F.; Filho, E.F.S. Simulation of an Artificial Hearing Module for an Assistive Robot. *Adv. Intell. Syst. Comput.* **2019**, *1*, 852–865.
3. Martins, P.S.; Faria, G.; Cerqueira, J.J.F. I2E: A Cognitive Architecture Based on Emotions for Assistive Robotics Applications. *Electronics* **2020**, *9*, 1590. [CrossRef]
4. Baek, J.Y.; Lee, S.P. Enhanced Speech Emotion Recognition Using DCGAN-Based Data Augmentation. *Electronics* **2023**, *12*, 3966. [CrossRef]
5. Khare, S.K.; Acharya, U.R. Adazd-Net: Automated adaptive and explainable Alzheimer's disease detection system using EEG signals. *Knowl.-Based Syst.* **2023**, *278*, 1–16. [CrossRef]
6. Nwe, T.L.; Foo, S.W.; De Silva, L.C. Speech emotion recognition using hidden Markov models. *Speech Commun.* **2003**, *41*, 603–623. [CrossRef]

7. Schuller, B.; Rigoll, G.; Lang, M. Hidden Markov model-based speech emotion recognition. In Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03), Hong Kong, China, 6–10 April 2003; Volume 2, pp. II–1.
8. Lanjewar, R.B.; Mathurkar, S.; Patel, N. Implementation and comparison of speech emotion recognition system using Gaussian Mixture Model (GMM) and K-Nearest Neighbor (K-NN) techniques. *Procedia Comput. Sci.* **2015**, *49*, 50–57. [CrossRef]
9. Utane, A.S.; Nalbalwar, S. Emotion recognition through speech using Gaussian mixture model and hidden Markov model. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2013**, *3*, 742–746.
10. Gupta, M.; Patel, T.; Mankad, S.H.; Vyas, T. Detecting emotions from human speech: Role of gender information. In Proceedings of the 2022 IEEE Region 10 Symposium (TENSYMP), Mumbai, India, 1–3 July 2022; pp. 1–6.
11. Kim, S.; Lee, S.P. A BiLSTM—Transformer and 2D CNN Architecture for Emotion Recognition from Speech. *Electronics* **2023**, *12*, 4034. [CrossRef]
12. de Lope, J.; Graña, M. An ongoing review of speech emotion recognition. *Neurocomputing* **2023**, *12*, 4034. [CrossRef]
13. Ottoni, A.L.C.; Souza, A.M.; Novo, M.S. Automated hyperparameter tuning for crack image classification with deep learning. *Soft Comput.* **2023**, *27*, 18383–18402. [CrossRef]
14. Ottoni, A.L.C.; de Amorim, R.M.; Novo, M.S.; Costa, D.B. Tuning of data augmentation hyperparameters in deep learning to building construction image classification with small datasets. *Int. J. Mach. Learn. Cybern.* **2023**, *14*, 171–186. [CrossRef] [PubMed]
15. Ottoni, L.T.C.; Cerqueira, J.J.F. Optimizing Speech Emotion Recognition: Evaluating Combinations of Databases, Data Augmentation, and Feature Extraction Methods. In Proceedings of the XVI Brazilian Congress on Computational Intelligence, Salvador, Brazil, 8–11 October 2023.
16. Mantovani, R.G.; Rossi, A.L.; Alcobaça, E.; Vanschoren, J.; de Carvalho, A.C. A meta-learning recommender system for hyperparameter tuning: Predicting when tuning improves SVM classifiers. *Inf. Sci.* **2019**, *501*, 193–221. [CrossRef]
17. Aguiar, G.J.; Santana, E.J.; de Carvalho, A.C.; Junior, S.B. Using meta-learning for multi-target regression. *Inf. Sci.* **2022**, *584*, 665–684. [CrossRef]
18. Khare, S.K.; Blanes-Vidal, V.; Nadimi, E.S.; Acharya, U.R. Emotion recognition and artificial intelligence: A systematic review (2014–2023) and research recommendations. *Inf. Fusion* **2023**, *102*, 102019. [CrossRef]
19. Brazdil, P.; van Rijn, J.N.; Soares, C.; Vanschoren, J. *Metalearning: Applications to Automated Machine Learning and Data Mining*; Springer Nature: Cham, Switzerland, 2022.
20. Reif, M.; Shafait, F.; Dengel, A. Meta-learning for evolutionary parameter optimization of classifiers. *Mach. Learn.* **2012**, *87*, 357–380. [CrossRef]
21. Gupta, M.; Chandra, S. Speech Emotion Recognition Using MFCC and Wide Residual Network. In Proceedings of the 2021 Thirteenth International Conference on Contemporary Computing (IC3-2021), Noida, India, 5–7 August 2021; pp. 320–327.
22. Ahmed, M.R.; Islam, S.; Islam, A.M.; Shatabda, S. An ensemble 1D-CNN-LSTM-GRU model with data augmentation for speech emotion recognition. *Expert Syst. Appl.* **2023**, *218*, 1–21.
23. Pan, S.T.; Wu, H.J. Performance Improvement of Speech Emotion Recognition Systems by Combining 1D CNN and LSTM with Data Augmentation. *Electronics* **2023**, *12*, 2436. [CrossRef]
24. Asiya, U.; Kiran, V. Speech Emotion Recognition-A Deep Learning Approach. In Proceedings of the 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), Palladam, India, 11–13 November 2021; pp. 867–871.
25. Bautista, J.L.; Lee, Y.K.; Shin, H.S. Speech emotion recognition based on parallel CNN-attention networks with multi-fold data augmentation. *Electronics* **2023**, *11*, 3935. [CrossRef]
26. Bhangale, K.; Kothandaraman, M. Speech emotion recognition based on multiple acoustic features and deep convolutional neural network. *Electronics* **2023**, *12*, 839. [CrossRef]
27. Chitre, N.; Bhorade, N.; Topale, P.; Ramteke, J.; Gajbhiye, C. Speech Emotion Recognition to assist Autistic Children. In Proceedings of the 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 9–11 May 2022; pp. 983–990.
28. Jothimani, S.; Premalatha, K. MFF-SAug: Multi feature fusion with spectrogram augmentation of speech emotion recognition using convolution neural network. *Chaos Solitons Fractals* **2022**, *162*, 112–512. [CrossRef]
29. Livingstone, S.R.; Russo, F.A. 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* **2018**, *13*, e0196391. [CrossRef]
30. Pichora-Fuller, M.K.; Dupuis, K. Toronto emotional speech set (TESS). *Sch. Portal Dataverse* **2020**, *1*, 2020.
31. Jackson, P.; Haq, S. *Surrey Audio-Visual Expressed Emotion (Savee) Database*; University of Surrey: Guildford, UK, 2014.
32. Cao, H.; Cooper, D.G.; Keutmann, M.K.; Gur, R.C.; Nenkova, A.; Verma, R. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE Trans. Affect. Comput.* **2014**, *5*, 377–390. [CrossRef] [PubMed]
33. Elgendy, M. *Deep Learning for Vision Systems*; Simon and Schuster: New York, NY, USA, 2020.
34. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **2011**, *12*, 2121–2159.
35. Dolka, H.; VM, A.X.; Juliet, S. Speech emotion recognition using ANN on MFCC features. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 431–435.

36. Ashok, A.; Pawlak, J.; Paplu, S.; Zafar, Z.; Berns, K. Paralinguistic Cues in Speech to Adapt Robot Behavior in Human-Robot Interaction. In Proceedings of the 2022 9th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob), Seoul, Republic of Korea, 21–24 August 2022; pp. 1–6.
37. Singh, J.; Saheer, L.B.; Faust, O. Speech Emotion Recognition Using Attention Model. *Int. J. Environ. Res. Public Health* **2023**, *20*, 5140. [CrossRef]
38. Nasim, A.S.; Chowdory, R.H.; Dey, A.; Das, A. Recognizing Speech Emotion Based on Acoustic Features Using Machine Learning. In Proceedings of the 2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 23–25 October 2021; pp. 1–7.
39. Hazra, S.K.; Ema, R.R.; Galib, S.M.; Kabir, S.; Adnan, N. Emotion recognition of human speech using deep learning method and MFCC features. *Radioelectron. Comput. Syst.* **2022**, *4*, 161–172. [CrossRef]
40. Lemke, C.; Budka, M.; Gabrys, B. Metalearning: A survey of trends and technologies. *Artif. Intell. Rev.* **2015**, *44*, 117–130. [CrossRef]