electronics

MDPI

Article

Environmental Sound Classification Based on Transfer-Learning Techniques with Multiple Optimizers

Asadulla Ashurov [1,2], Yi Zhou [1,2,*], Liming Shi [1,2], Yu Zhao [1,2] and Hongqing Liu [1,2]

[1] School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; asadullahashur@gmail.com (A.A.); shilm@cqupt.edu.cn (L.S.); zhaoyu@cqupt.edu.cn (Y.Z.); hongqingliu@cqupt.edu.cn (H.L.)
[2] Chongqing Key Laboratory of Signal and Information Processing, Chongqing 400065, China
* Correspondence: zhouy@cqupt.edu.cn

Abstract: The last decade has seen increased interest in environmental sound classification (ESC) due to the increased complexity and rich information of ambient sounds. The state-of-the-art methods for ESC are based on transfer learning paradigms that often utilize learned representations from common image-classification problems. This paper aims to determine the effectiveness of employing pre-trained convolutional neural networks (CNNs) for audio categorization and the feasibility of retraining. This study investigated various hyper-parameters and optimizers, such as optimal learning rate, epochs, and Adam, Adamax, and RMSprop optimizers for several pre-trained models, such as Inception, and VGG, ResNet, etc. Firstly, the raw sound signals were transferred into an image format (log-Mel spectrogram). Then, the selected pre-trained models were applied to the obtained spectrogram data. In addition, the effect of essential retraining factors on classification accuracy and processing time was investigated during CNN training. Various optimizers (such as Adam, Adamax, and RMSprop) and hyperparameters were utilized for evaluating the proposed method on the publicly accessible sound dataset UrbanSound8K. The proposed method achieves 97.25% and 95.5% accuracy on the provided dataset using the pre-trained DenseNet201 and the ResNet50V2 CNN models, respectively.

Keywords: environmental sound classification; transfer learning; CNN; pre-trained model

check for updates

Citation: Ashurov, A.; Zhou, Y.; Shi, L; Zhao, Y.; Liu, H. Environmental Sound Classification Based on Transfer-Learning Techniques with Multiple Optimizers. *Electronics* **2022**, *11*, 2279. https://doi.org/10.3390/electronics11152279

Academic Editor: George Angelos Papadopoulos

Received: 24 June 2022
Accepted: 12 July 2022
Published: 22 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Sound provides multidimensional information with various functions [1,2], and sound categorization has captured researchers' interests with various machine-learning and deep-learning (DL) algorithms and procedures being applied. Environmental sound classification (ESC) enables sound events that are frequently audible indoors or outdoors to be automatically detected and classified into various categories. Many competitions for machine learning, automation [3], robotics or assistance for the deaf and elderly [4,5] are driving research in this area to promote auditory machine awareness to a level comparable to that of human perception. Conventional machine-learning approaches (e.g., support vector machines and naive Bayes) and DL (e.g., convolutional neural networks (CNNs)) are examples of such processes. DL systems, especially those based on CNNs, have shown to be promising in various identification and classification tasks, most notably in image recognition [6,7]. Actually, the application field can be extended to sound-event classification after the basic sound information is converted and transformed to the resulting image form representations (such as spectrograms and scalograms) [8,9]. In addition, the creation of transfer-learning models using pre-trained weights, discriminative learning, and fine-tuning concepts has been proven. Numerous researchers developed their robust classification models using various transfer-learning strategies [10,11]. Hence, with the development of the DL techniques, it has been demonstrated that deep neural networks (DNNs) have a robust ability to extract features, and sound-classification tasks motivated an

increase in the application of deep network models to solve the ESC problem automatically. Notably, it has been demonstrated that the CNN allows models to overcome the limits of conventional approaches. CNN provides feature parameter sharing and dimension reduction [12]. The number of parameters in CNN is lowered due to parameter sharing. Consequently, computations are also reduced [13]. CNN can automatically extract the features from data and acquire scores based on the output [14]. Thus, it is regarded as highly ideal for tackling ESC tasks. CNN is efficient at not only detecting images; it is also suitable for spectrogram classification because it can acquire translation-invariant and spatially hierarchical patterns [15–17]. On the other hand, using a model that can be reused in various ways reduces the time and cost spent on training the whole network instead of only specific segments (usually those relevant to classification). This paper examines multiple CNN pre-trained models on the public environmental sounds dataset Urbansound8K [18], identifies the model that achieves the most remarkable performance as the best one for sound spectrogram classification, and refines it for better classification with a transfer-learning approach, and applies fine-tuning to the pre-trained models. The essential contributions of this study can be summarized as follows:

- We investigated pre-trained CNNs and utilized transfer learning to classify environmental sounds into ten categories accurately. Then, we compared the classification performance of several pre-trained models using the ambient sound dataset and analyzed the advantages and disadvantages of these strategies.
- We used a publicly accessible environmental sound dataset to test the effectiveness of transfer learning on the chosen CNNs regarding classification accuracy, precision, recall, and F1-score.
- In this study, nine commonly used pre-trained models are utilized in conjunction with a fine-tuning strategy to reduce training time while maintaining output accuracy.
- We assessed the performance of the modified pre-trained models, examined their benefits, and compared the obtained results with those from the related methods.

The rest of this paper is arranged as follows. Section 2 summarizes the works related to the ESC. Section 3 addresses the methodologies and materials used in this work, including transfer-learning, hyper-parameters and fine-tuning, and brief descriptions of modified pre-trained CNN models. Section 4 outlines the simulation procedures and results in terms of the experimental design, data-set description, and evaluation metrics for this study, and discusses the research findings. Section 5 draws some conclusions.

2. Related Works

In recent years, the quantity of work performed in the domain of ESC has increased exponentially. Numerous research has been undertaken to evaluate its applicability. Classifiers such as neural networks, support vector machines (SVM) [19], k-nearest neighbor (kNN) [19], and random forest (RF) [20,21] are used to provide accurate analysis results. Environmental sound categorization research is scarce compared to other machine learning sound- and image-processing challenges. Recent works have demonstrated this concept's utility in a variety of contexts, such as virtual assistants [22], automatic voice recognition [23], and text-to-speech applications [24]. The classifier employed in these investigations divides these recent works into two kinds. Traditional machine learning relies on customized input attributes. In the second approach, deep-learning-based classifiers learn features from input data. The following paragraphs discuss these two approaches to ESC systems.

2.1. Conventional Methods of ESC

Cowling and Sitte explored numerous classification strategies and discovered that Mel-frequency-cepstral-coefficients (MFCC)-based methods have the greatest classification rate of 70% [25]. Lu and colleagues used SVM for audio classification, claiming that higher classification accuracy and performance than kNN and other approaches were obtained [26]. Pillos and colleagues investigated the use of a combination of distinct audio-

feature extraction methods to improve the accuracy of ambient sound identification. They used a variety of feature extraction approaches, including zero-crossing rate and MFCC. To identify the sound, a multi-layer perceptron classifier was utilized. This approach achieved the highest accuracy of 74.5% on the chosen dataset [27], the same feature sets as MFCC, and obtained an accuracy of 88.02% for Urbansound8k [28]. Additionally, in [29], Uzkent, Barkana, and Cevikalp proposed a novel two-dimensional feature set based on the pitch range (PR) and MFCC of environmental sounds, as well as an autocorrelation function for feature extraction through SVM. Among the classifiers they employed, the SVM classifier with the Gaussian kernel had the best accuracy of 85.6% in recognizing nonspeech ambient sounds. Accordingly, it can be concluded that conventional machine-learning models do not consistently outperform or function as predicted when used for ESC [30]. CNNs are more powerful and effective than machine-learning algorithms when a considerable quantity of data can be learned and high-level features can be extracted from complex datasets [31]. The goal is to enhance the performance of sound-processing models that can be utilized with pre-processed sound datasets—identifying ambient sounds using CNN with a transfer learning approach and pre-trained models. It provides a solution by emphasizing the impact on the performance of sound classification based on the input spectrograms.

2.2. Spectrogram-Based ESC with CNN

Piczak evaluated the effectiveness of a CNN model in the categorization of environmental sounds, training the model using segmented spectrograms and evaluating its output. Compared to conventional techniques, this method attained significant performance by extracting the log-Mel spectrogram from each frame as an audio feature using a CNN model [32]. Salamon and Bello [33] used various data-augmentation techniques on the audio recordings to improve the data quality and diversity, allowing them to train their deep CNN model more effectively. Their approach earned significantly higher classification accuracy on the Urbansound8k dataset. Zhou et al. studied ESC utilizing CNN input spectrograms techniques, achieving excellent results, with 86% accuracy in a 2048 frame length [34]. While Zhang and Zou [35] achieved 81.9% accuracy with their dilated CNN, their work did not include any additional optimizers except SGD, which we consider to be one of the study's shortcomings. In [9], the authors conducted research on accurate classification using image-recognition models such as GoogleNet and Alexnet, achieving exceptional results of 93% on the Urbansound8k dataset. Li et al. [36] suggested a hybrid CNN model for environmental sound detection, with the Ave-CNN and Pro-CNN models achieving 91.6% and 91.9% accuracy, respectively, on the Urbansound8k dataset. In [37], the author developed a method for classifying domestic multi-channel audio by comparing different combinations of pre-trained CNN models and SVM. The neural network model was initially trained using spectro-temporal features extracted from the audio, represented by scalogram images generated using the continuous wavelet transform (CWT). The result obtained in this study was an F1 score of over 97% for the Xception network and combined with the multi-class linear SVM for classification purposes. Zeng et al. [38] proposed a DNN-based multi-task model that exploits numerous audio-categorization tasks. The authors' approach is referred to as the gated residual networks (GResNets) model because it combines deep residual networks (ResNets) with a gate mechanism to extract better representations than CNNs. ResNets replace two feed-forward convolution layers with two multiplied convolutional layers. This experiment evaluated the proposed model on numerous audio-classification tasks and discovered that the suggested multi-task model achieves higher accuracy than task-specific models trained independently. In [39], the authors' proposed method encompassed multiple stages, including pre-processing, feature extraction based on deep learning, feature concatenation, feature reduction, and classification. The input sound signals are denoised and transformed into sound images using the short-time-Fourier-transform (STFT) technique. Once the audio images have been generated, the CNN models VGG16, VGG19, and DenseNet201 are employed to extract features. This investigation used a support vector machine (SVM) classifier to classify

the sound images. The experimental results indicate that the suggested method obtained 94.8%, 81.4%, and 78.14% accuracy scores for ESC-10, ESC-50, and UrbanSound8K datasets, respectively. As evidenced by the observation of the studies above, most researchers have employed many strategies and methodologies to enhance performance. Numerous authors have employed CNN. Some studies have been conducted. However, there remains space for performance improvements, reducing the training time and achieving better accuracy. Thus, for ESC challenges, developing a novel methodology that combines the impacts of regularization, a perfect audio feature-extraction technique, and suitable CNN models are required to achieve high accuracy. This study proposes feature extraction and classification techniques based on transfer learning with several pre-trained CNN models.

3. Methodologies and Materials

We propose to use transfer-learning techniques for classifying ambient sounds with different fine-tuned CNN models. The framework of the transfer-learning technique employed in this investigation is depicted in Figure 1. The proposed approach in this research entails transforming sound clips into log Mel-spectrogram images and successfully using transfer-learning algorithms for ESC, in which several optimizers are applied. For the training process using the transfer-learning algorithms, we selected nine commonly used pre-trained models from the Keras library. As for the training and testing the dataset, it was created by converting raw audio files of Urbansound8K [18] to spectrogram images. The benefit of this dataset is that it enables experimentation with multiple-class classification. Indeed, this dataset encompasses ten distinct categories. Initially, pre-processing methods were applied to the target dataset. The proposed technique's performance was evaluated on the UrbanSound8K ESC dataset. The input sound clips were pre-processed to remove noises. The denoised sound signals were converted into spectrogram images. The STFT approach was utilized for converting sound clips into log Mel-spectrogram images. Then, data normalization was applied to rescale the pixel value of spectrogram images to the interval [0, 1]. Before feeding the spectrogram images to each network, the spectrogram images were adapted to each network's input dimensions using downsampling. After spectrogram images were created of the proposed strategy, an ESC classifier based on selected pre-trained models was utilized. The initial training level was the base model, containing well-known transfer-learning models such as Inceptionv3 [40], EfficientNetB0 [41], VGG19 [42], MobileNetV2 [43], DensNet201 [44], ResNet152V2 [45], ResNet50V2 [46], InceptionResNetV2 [47], and NASNetMobile [48]. Each pre-trained model was trained several times in this investigation using various optimizers, such as Adam, Adamax, and RMSprop, to obtain accurate results. After that, we defined the parameters for each transfer-learning network. The experimental results demonstrate that the proposed technique achieves slightly better classification accuracy. Additionally, the obtained results were compared to existing approaches and studies.

3.1. Transfer Learning

Transfer learning can be explained in terms of domains and tasks. Transfer learning [49] might be well described as taking knowledge from a source domain task and applying it to a target domain for improving the task. As a result, transfer learning can first address the most severe problem: a lack of well-labeled training data. Moreover, since previously acquired knowledge from various domains and activities can be reused, training a model can significantly reduce time and computational resources. In statistics, the two elements of a tuple called a domain D is the feature space X and the marginal probability distribution $P(X)$, respectively. As a result, the domain can be represented as:

$$\mathcal{D} = \{\mathcal{X}, P(X)\}. \quad (1)$$

- Feature space: $\mathcal{X}$,
- Marginal distribution: $P(X), X = \{x_1, \cdots x_n\}, x_i \in \mathcal{X}$.

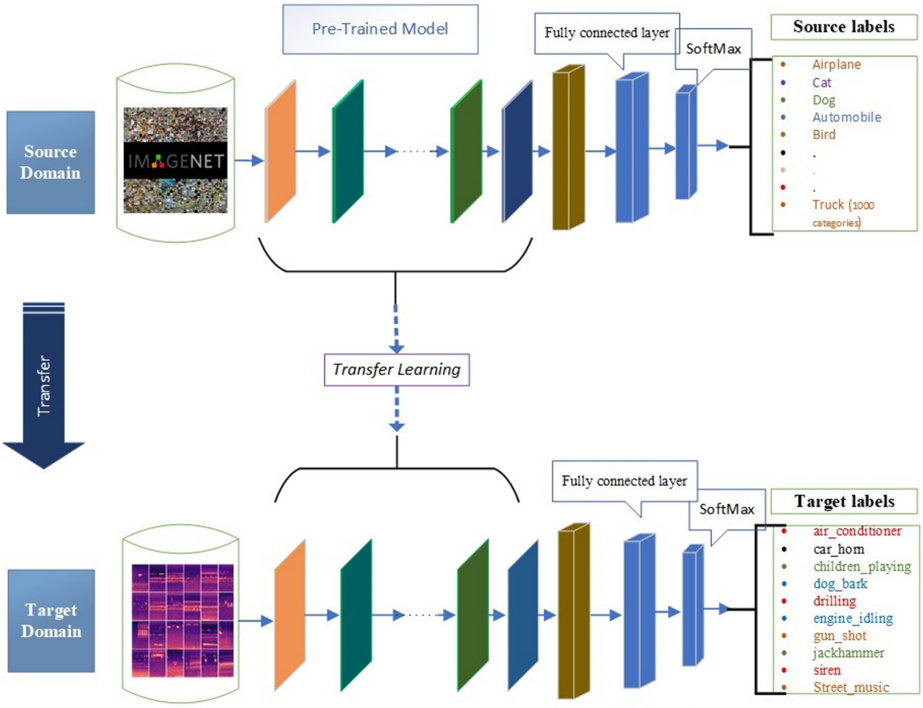

Figure 1. Illustration of transfer-learning workflow for environmental sound classification.

The overall design of the transfer-learning technique is shown in Figure 1. Transfer-learning models have already been applied to images of a number of different items in the ImageNet dataset, which have been classified into 1000 classes and preserved for use in a variety of other similar situations. Nowadays, transfer-learning methods are quite widespread and used in a wide variety of sectors [50–53]. The same technology as described above was used in this study on the selected audio dataset of environmental sounds. As seen in Figure 2, all modified pre-trained models that have applied the transfer-learning method are illustrated, which were utilized for this study.

3.2. Fine Tuning

Fine tuning is a well-known transfer learning strategy for neural networks in which a few training rounds are applied to the parameters of a pre-trained model to adapt them to new tasks. By incorporating data from an existing neural network and utilizing it as a starting point, fine tuning methods can enhance the accuracy of a new neural network model and make the training process more time and resource efficient. Fine tuning is a process for making incremental tweaks to obtain desired results. A prior DL algorithm's weights are used to design another matching DL algorithm. Keras, TensorFlow, and Torch are popular fine tuning frameworks. This study utilized the high-level Keras application programming interface (API) of Tensorflow for transfer learning and fine tuning workflows for pre-trained models. When training with pre-trained models on the Urbansound8K dataset, fine tuning is applicable for all pre-trained models. While the early layers of a CNN include available features, the later levels contain more specialized features to the original dataset's classes. In order to adapt the pre-trained models to the new classification task, the last original fully connected layer associated with the ImageNet classification task with 1000 classes is replaced with a series of layers that adapt the model to the new task to 10 classes. The deep neural network model can be employed successfully in new classification tasks when the parameters of the early layers are kept and those of the later layers are modified during training. As a result, fine tuning improves the performance and accuracy of the current classification process by adjusting parameters acquired from

prior network training on an extensive dataset. Utilizing data from a comparable existing structural CNN network can save significant time and cost. Thus, this paper proposed a transfer-learning-based strategy with fine tuning. This study used pre-trained models for classifying environmental sounds with spectrogram images. The particular endeavor consists of two stages. The initial stage is to freeze the top layers of the pre-trained models, after which five additional layers are added to the networks with the fine tuning process. The 5 more layers consist of an average pooling layer with a pool size of 3×3, a flattened layer, a dense layer with 128 neurons, a ReLU activation function, and a dropout rate of 0.4. The batch normalization technique is implemented to make learning easier and speed up training. Then, a decisive dense layer comprised of ten neurons and a Softmax activation function was added to classify environmental sounds using spectrogram images. In addition, three optimizer algorithms (Adam, RMSprop, and Adamax) were investigated to optimize the loss function during the process of fine-tuning.

3.3. Modified Pre-Trained Models

In this study, the transfer learning method is applied to the categorized spectrogram images with the pre-trained networks, which were trained on the ImageNet dataset, enabling the network to classify and recognize environmental sounds spectrogram images with high accuracy. Figure 2 contains information on the proposed neural network models.

(A.) InceptionV3. The Inception model is a deep CNN architecture presented by Szegedy et al. in 2014 on ImageNet dataset [40]. It is desirable to reduce the impact of computing complexity and have low parameters in application circumstances. This model utilizes convolutional kernels of varying sizes, allowing it to have receptive fields with distinct surface areas. The Inception layer combines 1×1, 3×3 and 5×5 convolutional layers. The image dimensions entering into InceptionV3 are 299×299. In order to limit the network's architecture area, it employs a modular structure followed by final joining, so as to achieve the fusion of elements with varied sizes. In this model, the ReLU function and batch normalization were used for activation. ReLU function can be represented as follows:

$$y(H^l_{ijk}(t)) = \max\left(0, H^l_{ijk}(t)\right) \tag{2}$$

where i and j denote pixels of input sample on the k^{th} feature map, and $H^l_{ijk}(t)$ denotes the output of the network by time step t.

The InceptionV3 model employs several strategies to optimize the network for improved model adaptability. It has a more extensive network than the Inception V1 and V2 models, but its speed is unaffected. It is also less computationally complicated, since it uses auxiliary classifiers as regularizers. InceptionV3 is a well-known image-recognition model that has been shown to reach an accuracy of more than 78.1% on the ImageNet dataset. The model is the culmination of several notions investigated by numerous researchers [54]. The modified InceptionV3 model is illustrated in Figure 2a.

(B.) EfficientNetB0. EfficientNetB0 is a model from Tan and Le's 2019 family of neural network models, which are produced by randomly selecting the scaling factor of the models. Computational resources are lost if the resolutions are not divisible by 8 or 16, since they result in zero padding at the bounds, so the resolution selected for B0 is 224. The EfficientNet blocks' width and depth should have channel sizes of 8. When memory is limited, the resolution can be reduced, but the depth and breadth of the model can be increased to improve speed.

This study uses the EfficieNetB0 model with changed ending layers inserted by layer freezing via fine tuning to classify and recognize environmental sounds. This work compares the improved EfficentNetB0 to other current pre-trained models in classifying and recognizing environmental sounds, which has not been found in literature to our best knowledge. The modified EfficientNetB0 model is presented in Figure 2b.

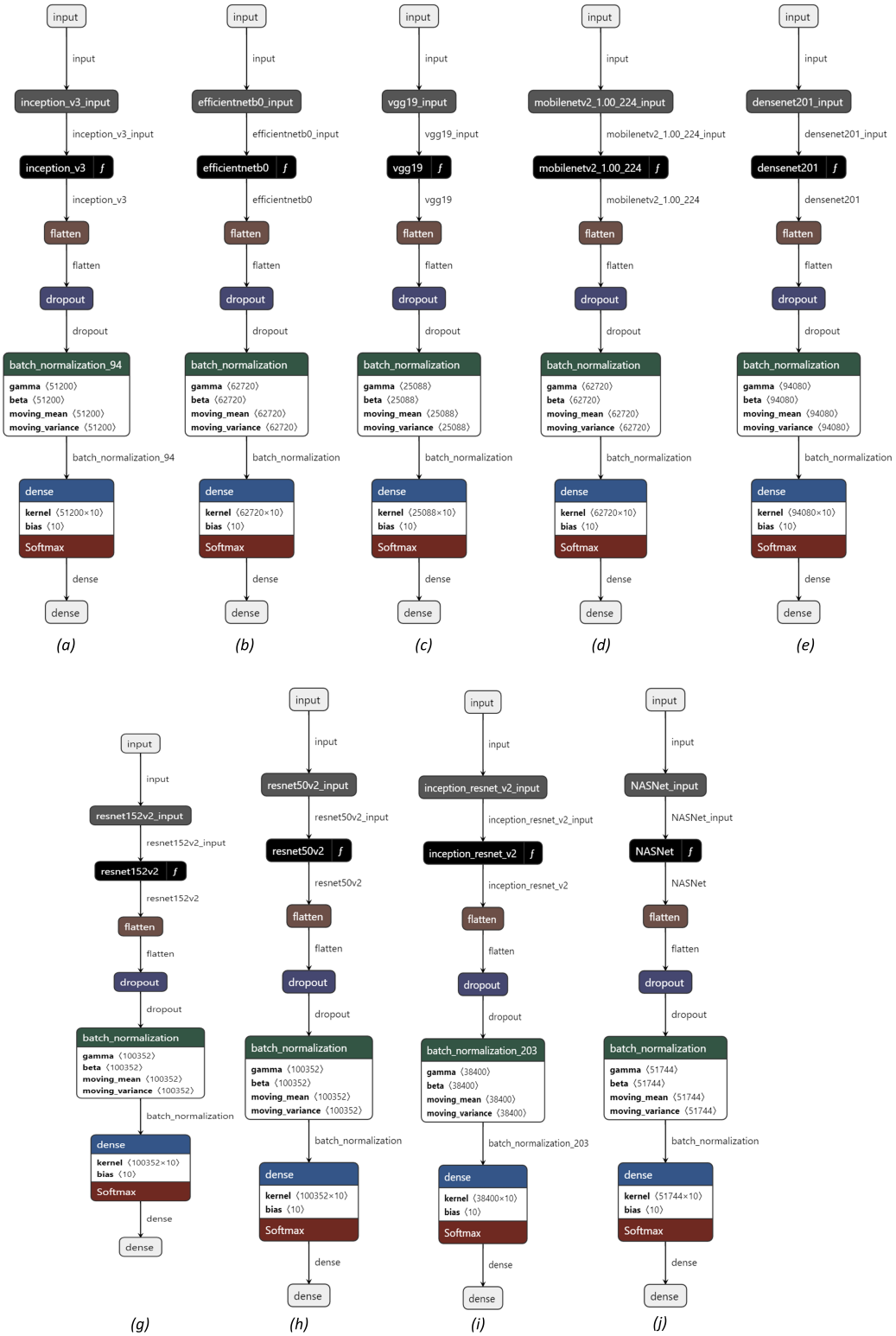

Figure 2. Modified blocks of pre-trained models after fine-tuning processes: (**a**)-InceptionV3, (**b**)-EfficientNetB0, (**c**)-VGG19, (**d**)-MobileNetV2, (**e**)-DenseNet201, (**g**)-ResNet152V2, (**h**)-ResNet50V2, (**i**)-InceptionResNetV2, and (**j**)-NasNetMobile. Where **f** refers to the basic pre-trained architecture in these models.

(C.) VGG19. VGG19 is a deep CNN model developed for image recognition based on the ImageNet dataset [55]. It consists of 19 layers with 16 convolution layers and 3 fully linked layers to categorize images into 1000 categories. The convolution layers are divided

into five groups, with a max-pooling layer after each group. Due to the employment of multiple 3 × 3 filters in each convolutional layer, it is the commonly used model for image classification. The VGG19 first layer has an input size of 224 × 224 × 3, whereas the first convolutional layer has 32,364 weights. On the first layer, the total learnable weights are 1792, whereas, on the second layer, they are 36,292. The object in the image's label is output by the model. This study utilizes the VGG19 transfer-learning model with changed last layers, which are classification parts inserted by freezing the initial layer via fine tuning to classify and recognize environmental sounds. Classification layer was applied with Softmax function and dense layer with 10 neurons. Figure 2c illustrates the improved VGG19 pre-trained model.

(D.) MobileNetV2. MobileNetV2 [43] improves the present performance of mobile models across a spectrum of workloads and benchmarks, as well as across a range of model sizes. It is a powerful feature extractor for object detection and classification. MobileNets are parameterized to be compact, low-latency, and low-power to fulfill the resource restrictions of various usages. The fundamental building component is a bottleneck depth-separable convolution with residuals. MobileNetV2's design begins with a 32-filter fully convolutionary layer, followed by 19 bottleneck-containing layers. The input image resolution and the width multiplier have been used as customizable hyperparameters to adapt the architecture to various performance factors [56]. In the primary network (width multiplier 1, 224, 224), 3.4 million parameters are employed, with a computational cost of 300 million multiply adds. The computational cost of the network ranges up to 585 M, and the model size spans from 1.7 M to 6.9 M parameters, with 2.1 M parameters trained in this experiment. This study applies fine tuning in this transfer-learning model by freezing the initial layer and adding a classifying layer. The modified MobileNetV2 pre-trained model is shown in Figure 2d.

(E.) DenseNet201. There are 201 layers in DenseNet-201's CNN [44]. It is trained on more than a million images from the ImageNet database. The pre-trained network can categorize images into 1000 item categories, various objects, things, and numerous images. Consequently, the network has learned rich feature representations from various images. The network can accept images up to 224 × 224 pixels in size. DenseNet-201 has accumulated a wealth of feature representations from various images. Besides the improved parameter efficiency, a significant benefit of DenseNet is the greater flow of information and gradient, which simplifies training images throughout the network. This study optimized the DenseNet201 pre-trained model for multi-class ESC. The input scale for this transfer-learning model is 224 dimensions. Therefore, spectrogram images are downsampled to feed this model. Its classification parts are applied with Softmax function and dense layer with 10 neurons and the top layer is frozen to classify environmental sounds using spectrogram images. The classifier levels are removed, and a new fully connected layer with 10 different environmental sounds is added. The precise weights were employed in the fine-tuned model, and transfer learning was used to train this model. This transfer learning model is modified following the fine-tuning technique discussed in the previous section. Figure 2e illustrates the improved DenseNet201 model.

(G.) ResNet152V2. A residual network (ResNet) has a CNN architecture with hundreds or thousands of convolutional layers used in computer vision. Previous CNN structure limits the efficacy of succeeding layers. ResNet is made up of a huge number of layers that work very well. The primary difference between ResNetV2 [46] and its first version is that it performs batch normalization on each weight layer before training, while its first version does not. ResNet performs well in image identification and localization tasks, illustrating the importance of a wide variety of visual-recognition challenges. The classifier part of the ResNet152V2 pre-trained model consists of five layers, 3 × 3 average pooling layer, flatten layer, and dropout with 0.4. The last layer is the soft-max layer, which outputs 10 class values. Figure 2g depicts the improved ResNet152V2 pre-trained model.

(H.) ResNet50V2. Deeper CNNs, mostly, need more training time. As the depth of the layer increases, the model's accuracy frequently decreases, including network optimization,

the vanishing gradient problem, and degradation issues. ResNet50, on the contrary, is easier to modify and it can attain high accuracy as the network depth increases. There are 50 layers in this model, used 3-layer bottleneck blocks to ensure improved accuracy and lesser training time. In ResNet50V2, the propagation formulation of the connections between blocks was improved. [46]. ResNet50 is a variant of ResNet with 48 convolutional layers, 1 MaxPool layer, and 1 Average Pool layer. It is capable of performing floating-point computations up to 3.8×10^9. It is a frequently used ResNet model, and we extensively examined the ResNet50V2 architecture.The classifier part of fine-tuned ResNet50 involves the average pooling function to reduce the number of parameters, dropout with 0.4 rates to avoid overfitting problems, and a fully connected layer that contains 10 nodes as the number of classes. The modified ResNet50V2 pre-trained model is shown in Figure 2h.

(I.) InceptionResNetV2. The InceptionResNetV2 model was trained using over a million images from the ImageNet dataset. This network has 164 layers and is capable of classifying images into 1000 categories using a variety of image formats.The network has an image dimension (input size) of 299×299 [40,57]. It is a more expensive hybrid version of Inception that provides much increased recognition performance [47]. The classifier part of InceptionResNetV2 is similar to ResNet50 and has an average pooling and fully connected layer. The final layer contains the Softmax function and Dense layer with 10 neurons. The modified InceptionResNetV2 model is shown in Figure 2i.

(J.) NasNetMobile. The term NASNet refers to a neural architecture search (NAS) network, a machine-learning model. The NasNetmobile architecture accepts 224×224 images as input [58]. Google Brain built the NasNetmobile architecture to establish another search area to facilitate transferability. NasNet design is based on reinforcement learning and is hyperparameter-dependent. The NASNet model's architecture produces higher outcomes with smaller model sizes and less complexity. NasnetMobile consists of 12 blocks, with 5.3 million parameters. The classifier part of fine-tuned NasNetMobile model involves a dropout with 0.4 rates to avoid overfitting problems and a fully connected layer that contains 10 nodes as the number of classes. Classifier architecture is similar to that discussed in fine-tuning section. The modified NasNetMobile model is shown in Figure 2j.

3.4. Hyperparameters

Selecting hyperparameters are the most distinguishing features of DL models. They will directly affect various factors such as memory and computational complexity. The performance of the applied approaches can be controlled by adjusting hyperparameters. In this study, some additional hyperparameters were selected with a specific technique. In addition, optimizers are also crucial in the area of DL, and this study examines several types of optimizers, such as Adam, RMSProp, and Adamax. More efforts are required to fine tune the hyperparameters to achieve effectiveness in most cases. While training the DL model, we must modify the weights for each epoch and minimize the error. Pre-trained models were tuned using the following hyperparameters, 0.0001 learning rate, 30 epochs, and 32, 64 batch sizes. In order to find optimal hyperparameters, each pre-trained model was tested 15 to 20 times with different values such as 10, 20, 25, and 30 epochs, and 0.0015, 0.001, and 0.0001 learning rates. Moreover, SGD and Adadelta optimizers were also tested. However, their findings are not included in this work because transfer-learning models did not achieve better results when applied to spectrogram data with those values and optimizers. Every training procedure was implemented by restarting devices to avoid bias problems and memorizing the previous model's information. If a gradient explosion happened, early stopping was implemented. In every model considered, the dropout rates were 0.4. Adam, RMSProp, and Adamax were, respectively, included as optimizers. In each convolutional layer, rectified linear unit (ReLU) activation function was implemented, transforming the weighted input sum into the node's output. As a consequence, hyperparameter tuning helped reduce overall loss and improved accuracy. A well-chosen set of hyperparameters can significantly increase the likelihood that a model will achieve better metric values such as accuracy, F1-score, and precision. Otherwise, it

may result in an endless cycle of ongoing training and optimization. Adaptive optimization techniques, such as Adam, RMSprop, and Adamax, are among the most effective optimizers. Since these optimizers estimate the gradient using internal model feedback, they are almost parameter-free [59,60]. In this study, the aforementioned adaptive optimization strategies are used to improve pre-trained models' training and validation accuracy.

4. Simulation and Results

4.1. Experimental Setup

Dataset description. We utilized the Urbansound8K dataset to conduct these experiments [18]. It comprises 8732 labeled sound samples from ten different classes, each lasting for four seconds. Each clip is derived from field recordings made available at www.freesound.org (accessed on 3 November 2014). The audio data had already been split, excerpted, and assigned to ten distinct folds. We investigated the class distributions of each fold to assess the dataset's balance; the bar graph in Figure 3 reveals that the dataset's class names are relevant to the number of samples. This approach of imbalance analysis indicates that there are two classes with somewhat fewer entries than the other eight classes. As a result, it does not seem to be much out of balance. After that data-balance assessment, we extracted raw audio samples into log- Mel-spectrograms, as shown in Figure 4, to feed the chosen pre-trained CNN models. To prevent over-fitting issues, we used several data-augmentation strategies. Various data-augmentation strategies have already been used to solve particular challenges in the area of data augmentation [61]. This study used the Keras API to execute the spectrogram image-augmentation techniques. Increasing the number of data samples by utilizing just the information in this data is optimal for reducing model over-fitting. Therefore, to avoid over-fitting problems, we applied various data-augmentation techniques, such as flipping spectrograms horizontally and vertically [62], rotating 30 degrees [63], zooming to a 0.3 range [64], and shifting width by 0.3 [65] to enhance the data in this study. From 8732 original spectrogram images, 43,660 spectrogram images were generated. Spectrogram images were downsampled before feeding to the pre-trained networks to obtain the dimensions of 299 × 299 × 3 for InceptionV3, InceptionResnetV2, and 224 × 224 × 3 for other pre-trained models utilized in this study.

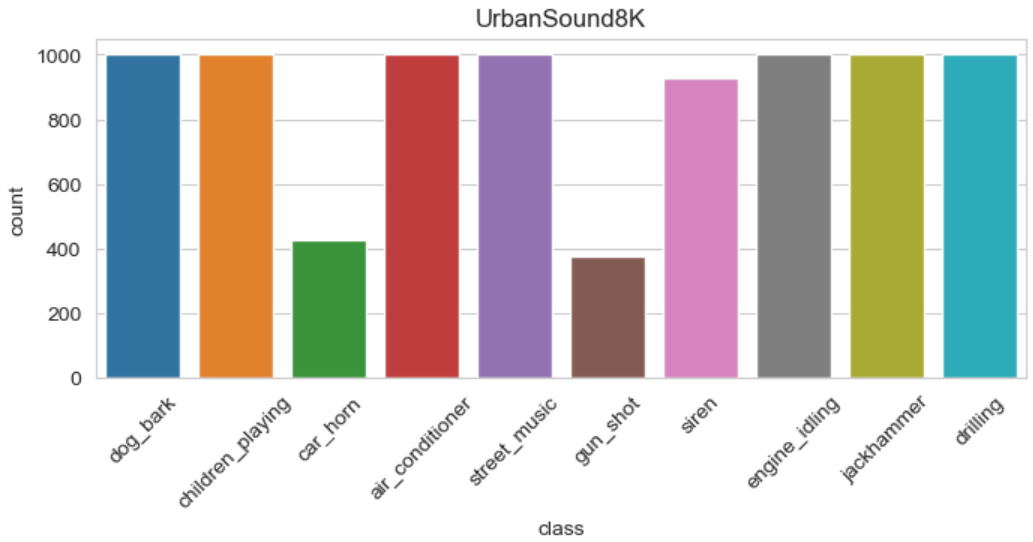

Figure 3. Urbansound8K dataset visualization.

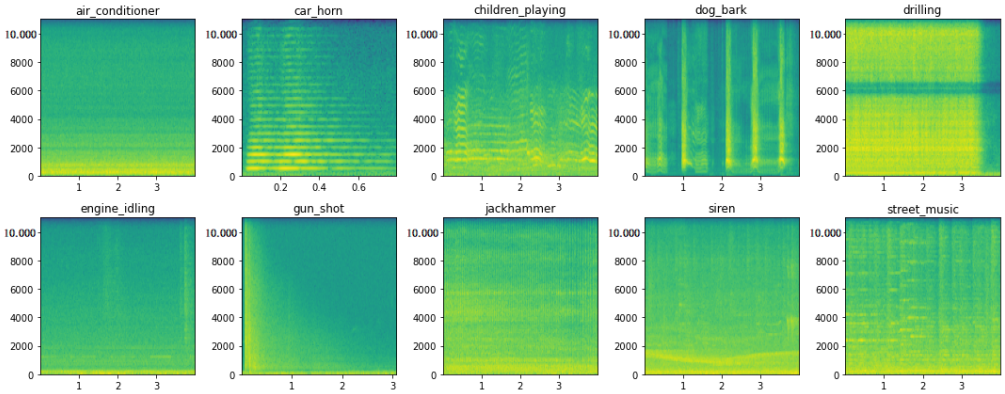

Figure 4. Illustrations of the log-Mel-Spectrogram samples from each class of the UrbanSound8K dataset.

Spectrogram; log Mel-spectrogram. A spectrogram represents sound as a frequency spectrum of an audio stream evolving with time. Typically, the sound is depicted as a wave shape, which is a two-dimensional representation of the amplitude and time components. The fast Fourier transform (FFT) [66] is used to generate a sound spectrogram from a temporal signal. Spectrograms are two-dimensional graphs, with a third dimension being represented by colors. The third dimension represents the amplitude of a specific frequency at a particular moment. Figure 4 illustrates the sample spectrograms from each of the ten classes of the selected dataset used in this study.

A spectrogram visually depicts a signal's frequency content across time [67,68]. Hence, Mel spectrograms are produced by applying a Mel filter bank on spectrograms obtained from audio signals using the STFT.

$$F(l, f) = \sum_{t=-\infty}^{\infty} m[t]\psi[t-l]e^{-jft} \tag{3}$$

where l represents the magnitude of the shift. f indicates the frequency and e^{-jft} denotes complex exponential. The Fourier transform is applied to contiguous segments of the signal $m[t]$ multiplied by the window function ψ[t]. STFT produces an array with each column representing the spectrum of the brief segment of the original signal. The Mel scale is a linear scale that corresponds to the human auditory system and is connected to Hertz through the following formula:

$$Mels = 1127ln_{10}(1 + Hertz/700) = 2595log_{10}(1 + Hertz/700). \tag{4}$$

The Mel spectrogram is used to provide the models with ambient sounds that are equivalent to what a human being would perceive [69].

Evaluation Metrics. This section discusses how to further examine and analyze the proposed models and methodologies. Numerous scientific studies use a range of evaluation methodologies, including the frequently used procedures for audio classification. In particular, [70] examined model-assessment strategies for data classification. Classification accuracy and other pertinent metrics can assess a model's performance in multi-class classification tasks. The following equations represent these metrics' mathematical formulations:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} * 100\%, \tag{5}$$

$$\text{Recall or Sensitivity } = \frac{TP}{TP + FN} * 100\%, \tag{6}$$

$$\text{F1-score} = \frac{2 * TP}{2 * TP + FP + FN} = 2 * \frac{\text{Precision } * \text{ Recall}}{\text{Precision } + \text{ Recall}} * 100\%, \tag{7}$$

$$\text{Precision} = \frac{TP}{TP + FP} * 100\%. \quad (8)$$

4.2. Results and Discussion

In this study, all pre-trained CNN models were fine-tuned and applied transfer learning; the knowledge from the original models were transferred to the target models, generating new adapted CNN models. This situation was similar to all pre-trained models we employed in this experiment. In this study, 80% of the datasets were used for training, and 20% were kept for validation. None of the validation dataset's instances appeared in the training dataset. Table 1 contains a detailed list of the parameters utilized in this investigation.

Table 1. Training parameters used in transfer-learning models in this study.

Parameters	Value
Number of training epochs	30
Batch size	32–64
Dropout	0.4
Learning rate	0.0001

Then, all pre-trained models were retrained using the supplied dataset. A slightly higher accuracy was achieved by using the 'Adam' optimizer with DenseNet201, ResNet50V2, and Inception-V3 models, respectively, 97.25%, 95.5%, and 92.5%. Additionally, inferior results were obtained with EfficientNetB0 when training with optimizers 'Adamax', 'RMSprop', and 'Adam'. Furthermore, the VGG-19 model being trained with 'RMSprop' and 'Adamax' illustrates low accuracy. The detailed results are summarized in Table 2. The subsequent paragraphs will discuss all results and findings in further depth.

Table 2. The performance comparison of metrics using transfer-learning models with different optimizers.

N	Pre-Trained Models	Adam		RMSprop		Adamax	
		Accuracy (%)	Loss (%)	Accuracy (%)	Loss (%)	Accuracy (%)	Loss (%)
1	InceptionV3	92.5	17.32	69.75	11.82	85.27	32.62
2	EfficientNetB0	32	46.22	35.75	65.5	41	65.25
3	VGG19	77.5	24.47	38.92	36.15	64.45	39.42
4	MobileNetV2	90.25	33.46	69.2	33.61	81.25	27.5
5	DenseNet201	97.25	12.61	95.5	13.5	93.75	19.25
6	ResNet152V2	89.67	18.25	55.45	42.28	76.75	26.5
7	ResNet50V2	95.5	16.41	85.5	17.37	75.25	28.45
8	InceptionResNetV2	91.25	13.71	92.64	23.46	69.67	38.12
9	NASNetMobile	79.25	22.5	87	26.5	70.5	36.75

Figure 5 illustrates the comparison of these transfer-learning models with the Adam optimizer on the Urbansound8k dataset. A clear distinction between the models' measures can be seen.

The comparison of the most frequently used assessment measures in transfer-learning models is shown in Table 2. A comparison of the accuracy of whole CNN pre-trained models using different optimizers with higher accuracy for various ESC is given in Table 2. The EfficientNetB0 model does not learn well with any of the three optimizers used in this study. It indicates 32%, 35.75%, and 41% accuracy, respectively. Additionally, this model indicates the poorest precision, of 38.31%, 27.3%, and 37.45% with the 'Adamax', 'RM-

Sprop', and 'Adam' optimizers. As discovered earlier, it can be concluded that this model is insufficiently adequate for this experimental analysis. In this scenario, EfficientNets has significantly lower performance in many aspects, such as accuracy, F1-score, and others, because EfficientNets has much less computational complexity and significantly greater data flow than comparable networks. Hardware accelerators, such as GPUs, can be employed for considerable computation, and data transport is a relatively minor performance factor. As a result, EfficientNetB0 underperforms on training operations, as EfficientNets are already sparse neural networks [71].

VGG19 has a total of 19 layers. It also outperforms on a range of tasks and datasets other than ImageNet. VGG19 is still one of the most widely used image-recognition architectures, although its performance is insufficient for our problem. It achieves a somewhat highest accuracy of 77.5% when being trained with the 'Adam' optimizer, shown in Figure 5. Furthermore, Figures 6 and 7 illustrate that the VGG19 model achieved 38.92%, and 64.45% accuracy when being trained with the 'RMSprop' and 'Adamax' optimizer. The precision and F1-scores for the VGG19 model are detailed in Table 3.

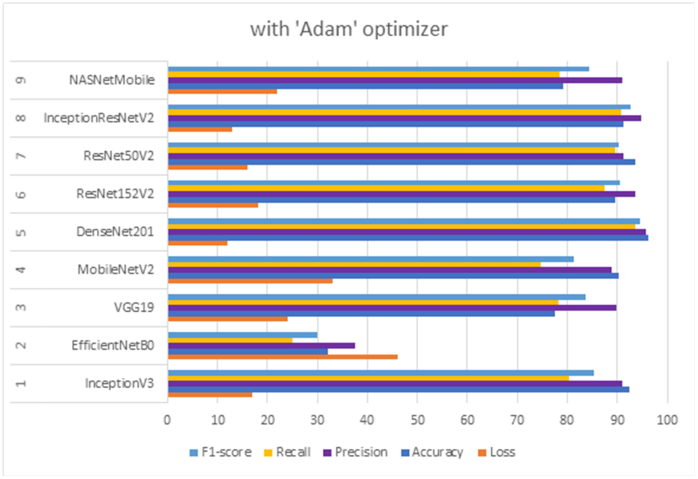

Figure 5. Comparative analysis of classification accuracy and loss obtained using transfer-learning techniques with Adam optimizer.

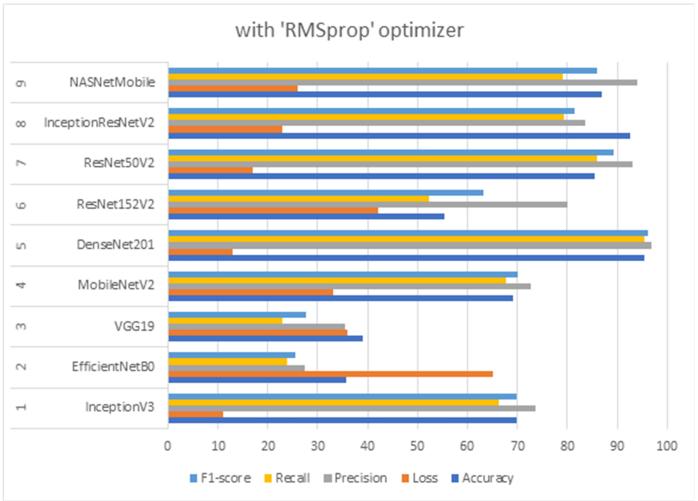

Figure 6. Comparative analysis of classification metrics obtained using transfer-learning techniques with RMSprop optimizer.

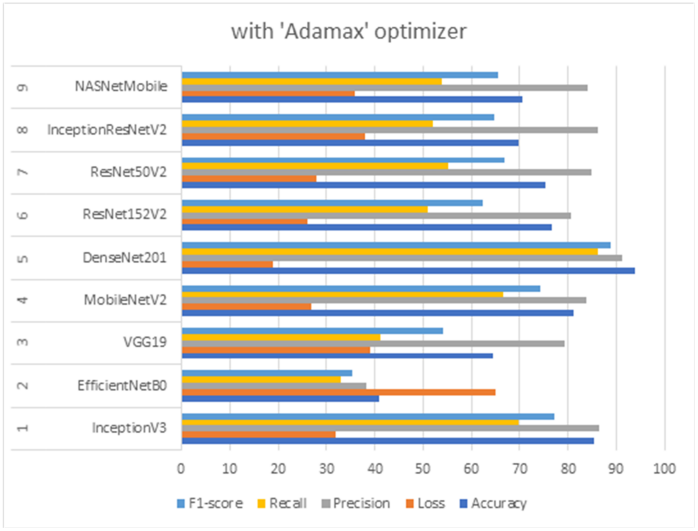

Figure 7. Comparative analysis of classification metrics obtained using transfer-learning techniques with Adamax optimizer.

From the results, it can be also seen that the 'Adam' optimizer performed well with MobileNetV2 in terms of line 4 in Figure 5, while RMSprop and Adamax achieved only 69.2% and 81.25% accuracy, respectively, in terms of line 4 in Figures 6 and 7. However, MobileNetV2 performs worse when being trained with the 'RMSprop' optimizer, achieving 69.2% accuracy and a higher validation loss of 33.61%. As shown in Table 3, MobileNetV2 model precision is 72.78%, and its F-score achieves 70.17%. In addition, other detailed findings are illustrated in Table 3.

Indeed, MobileNetV2 employs lightweight depth-wise convolutions to filter intermediate expansion layer information. While most researchers assert that it is a well-performing pre-trained model for most issues due to its lower computational complexity and other advantages, this model does not perform as well as expected in this study. It is not adaptive to ESC. In order to improve the accuracy of this transfer-learning model, it was upgraded with fine tuning and examined more than 15 times with three kinds of optimizers, which are used in this study. However, it cannot perform well in this experiment.

After being trained, the DenseNet201 network can produce excellent results despite its low computational complexity, which explains why this model requires a slightly longer training time than other chosen models. DenseNet201 provided various benefits that address the issue of vanishing gradients while simultaneously boosting parameters and computational efficiency, strengthening feature propagation, and encouraging feature reuse. DenseNet201 achieves significantly better results with all selected optimizers and learning rates than other utilized pre-trained models in this study. The results of classification accuracy through different optimizers using spectrogram images with CNN models for the chosen dataset are detailed in Table 2.

Residual neural networks achieve high-performance results by using skip connections to skip layers. Most ResNet models use double or triple layer skips with nonlinearities (ReLU) and batch normalization. Skip connections are used to prevent fading gradients or to reduce degradation. Figure 2g,h illustrate two residual network models that are used in this study. As shown in Table 2, ResNet152V2 and ResNet50V2 both perform well on the selected dataset, achieving excellent accuracy of 89.67%, 95.5% with the 'Adam', 55.45%, 85.5% with the 'RMSprop', and 76.75%, 75.25% with the 'Adamax' optimizer, respectively. Both of the residual network models have comparable accuracies, except that, when being trained with the 'RMSprop' optimizer, they differ by 32%. When comparing the training times of these models, a significant difference can be seen: ResNet152V2 needs much more training time than ResNet50V2. A comparison of the training times of these models is shown in Figure 8.

Table 3. Comparative analysis of classification metrics obtained using transfer-learning techniques with several optimizers.

		Adam		RMSprop		Adamax	
N	**Utilized Models**	**Precision**	**F1-Score**	**Precision**	**F1-Score**	**Precision**	**F1-Score**
1	InceptionV3	90.92	85.25	73.61	69.73	86.43	77.19
2	EfficientNetB0	37.45	29.98	27.3	25.40	38.31	35.45
3	VGG19	89.75	83.60	35.42	27.72	79.21	54.24
4	MobileNetV2	88.94	81.23	72.78	70.17	83.76	74.14
5	DenseNet201	95.65	94.56	96.84	96.16	91.33	88.72
6	ResNet152V2	93.62	90.46	80	63.21	80.59	62.47
7	ResNet50V2	91.3	90.39	93.02	89.37	84.74	66.89
8	InceptionResNetV2	94.87	92.76	83.65	81.39	86.17	64.86
9	NASNetMobile	91.11	84.34	94	85.84	84.19	65.61

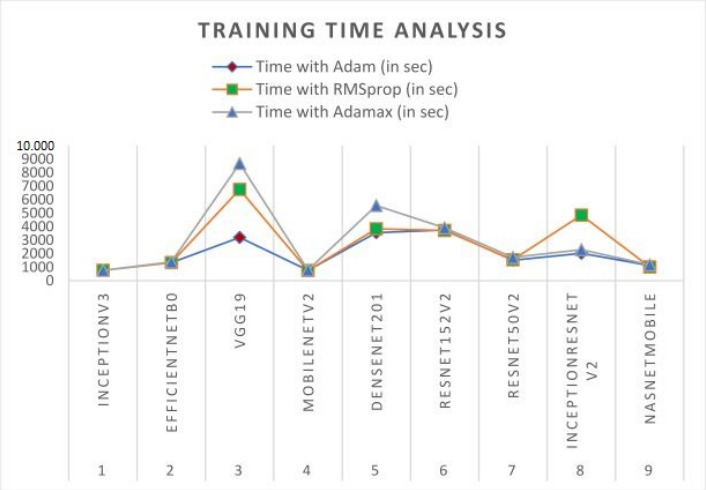

Figure 8. Computation time analysis of the pre-trained models.

Moreover, classifying ambient sounds using spectrograms is quite successful when using InceptionResNetV2. However, we must evaluate and decide which optimizers to use and tune their hyperparameters to increase the models' efficiency. It could be found that this pre-trained model performs excellently on the chosen dataset and achieves better accuracies with 'Adam' and 'RMSprop' optimizers, reaching 91.25% and 92.64%, respectively, in Figures 5 and 6. However, it obtains slightly weak accuracy with 'Adamax' optimizer, a 69.67% accuracy, as shown in Figure 7 and, when it comes to time consumption, it is an average pre-trained model for this problem.

NasNetMobile is a lightweight and well-known model for addressing most issues. This pre-trained model was used in this experiment to classify ambient sounds using spectrograms, and it was discovered that it performs well when being trained with the 'RMSprop' optimizer, reaching 87% accuracy, 94% precision, and an 85.84% F1-score, as shown in Table 3. However, when using the 'Adam' and 'Adamax' optimizers, the accuracy attained was 79.25% and 70.5%, respectively. By examining the time consumption of this pre-trained model, it can be concluded that it produces satisfactory results. Detailed information on time is shown in Figure 8.

The DenseNet201 model is quite effective and perfectly trained in this study, with a 97.25% accuracy that is 2.25% higher than the ResNet50V2, but this model takes significantly

less training time than DenseNet201, as shown in Figure 8. As a result, it can be concluded that this model is extremely suitable and effective for classifying ambient sounds and similar tasks. As shown in Tables 2 and 3, the InceptionV3 achieves excellent accuracy, precision, and F1-scores of 92.5%, 90.92%, and 85.25%, respectively, when being applied with 'Adam' optimizer.

When examining the precision of these models, it can be found that they are more precise than other pre-trained models, which indicates that the models work well, and this study recommends these pre-trained models for ESC problems. On the other hand, the EfficientNetB0 model's precision is not as high as expected, achieving just a maximum 41% accuracy with the 'Adam' optimizer. In addition, validation loss is very high when examined with all three optimizers, as shown in Figure 7. This model is not recommended for categorizing this kind of experiment. As a result, the F1-score also accurately indicates equivalent quantities in this study. All results of the F1-score are summarized in Table 3.

'RMSProp' is a related optimization technique to 'Adam'. However, 'RMSProp' with momentum is distinct from 'Adam' in several significant ways, such as 'RMSProp' with momentum creating parameter updates using momentum on the rescaled gradient. In contrast, 'Adam' directly updates using a running average of the first and second moments of the gradient. In addition, 'RMSProp' lacks a concept for bias correction [72]. In this study, these diversities have a substantial role in achieving greater performances in pre-trained models and some diversities between achieved results of the neural networks can be observed.

In this paper, nine commonly used pre-trained CNN architectures with transfer learning were employed for classifying the spectrogram images generated from ambient sounds into ten classes. Each model was assessed using the identical values for the epochs illustrated in Table 1. Table 2 demonstrates that DenseNet201 and ResNet50V2 perform better with the Adam optimizer among the nine transfer learning CNN models tested within this work. These models achieved 97.25% and 95.5% accuracies, respectively, with the Adam optimizer. The InceptionV3 and InceptionResNetV2 transfer learning models achieved 92.5% and 92.64% accuracy with Adam and RMSprop optimizers, respectively, as seen in Table 2. The performance metrics for these models are detailed in Tables 2 and 3. EfficientNetB0 achieved the lowest accuracy among the nine fine-tuned CNN models. It ranged from 32%, 35.75%, and 41% with Adam, RMSprop, and Adamax optimizers, respectively. Tables 2 and 3 depict the evaluation-performance metrics related to EfficientNetB0. The nine transfer-learning models were successfully applied to spectrogram images generated from ambient sounds, and most models performed much better. Consequently, it is essential to note that this research shows that some classification models, such as DenseNet and ResNet, are superior for classifying spectrogram images.

4.3. Comparison with Other Existing Published Studies

Finally, Table 4 compares this study's findings and achievements obtained via the transfer-learning technique with some earlier research results. For example, the work of Salamon compared a baseline system to the traditional classification method, achieving 66% accuracy. Dai and Pickzak utilized similar methods for ESC, and they obtained outcomes that are comparable to one another. Their findings are 71.8% and 73.7%, respectively. Chen and Zhang attained 78% and 81.9% accuracy, respectively, using dilated CNNs. Moreover, the proposed instance-specific adapted Gaussian mixture models by Chandrakala is also significantly higher than the previously considered models' findings, and the categorization accuracy reached 90%. Boddapati categorized ambient sounds using spectrograms and GoogLeNet, achieving a 93% accuracy rate, which is 4.2% lower than that of our proposed model. The DenseNet201 pre-trained model performed the best and yielded the most significant results with all three optimizers utilized in this study, according to the findings of this research. It achieved the most remarkable accuracy of 97.25%. The 'Adam' optimizer performs well with all the pre-trained models, including DenseNet201 with the proposed transfer-learning technique with the Log-Mel spectrogram images.

Table 4. Comparison of the previous state-of-the-art and transfer-learning approaches.

Author & Ref.	Method	Accuracy
J. Salamon [18]	SVM classification	65%
W. Dai [73]	Deep CNNs with up to 34 weight layers	71.8%
K. J. Pickzak [32]	CNN	73.7%
Y. Chen [74]	Dilated Convolution	78%
X. Zhang [35]	Dilated-CNNs	81.9%
S. Chandrakala [75]	Instance-specific adapted Gaussian mixture models	85.47%
S. Li [36]	Dempster–Shafer CNNs	90.2%
V. Boddapati [9]	GoogLeNet using Spectrogram feature	93%
Our result	Transfer-learning approaches	97.25%

5. Conclusions

This study proposes a transfer-learning approach for ESC based on log Mel-spectrogram on the UrbanSound8K dataset, with ten classes. The pre-trained CNNs models performed very well in this study. Exploring the chosen dataset and various training settings can help determine the most successful combinations. Our classification technique represents a significant step forward in developing accurate classifiers for an increasingly complex environmental sound dataset, particularly those incorporating distinct spectrogram representations generated from wave-forms of sounds. Selected pre-trained models perform well and attained a high accuracy in this study. Regarding the future study, we are interested in adopting attention mechanisms with CNN since they are optimal, but they also suffer from less over effects on the spectrograms generated. Adopting attention mechanisms with CNN on the ESC tasks may potentially increase the classifier's performance even further.

Author Contributions: Conceptualization, A.A.; formal analysis, Y.Z. (Yi Zhou), L.S., Y.Z. (Yu Zhao) and H.L.; funding acquisition, Y.Z. (Yi Zhou); investigation, L.S.; methodology, A.A.; project administration, Y.Z. (Yi Zhou) and H.L.; software, A.A.; writing—original draft, A.A.; writing—review & editing, Y.Z. (Yi Zhou), L.S. and H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This study is supported by the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJQN201900605).

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

References

1. Cheng, L.; Ji, X.; Zhao, H.; Li, J.; Xu, W. Tensor-based basis function learning for three-dimensional sound speed fields. *J. Acoust. Soc. Am.* **2022**, *151*, 269–285. [CrossRef] [PubMed]
2. Dang, X.; Zhu, H.; Cheng, Q. Multiple Sound Source Localization Based on a Multi-Dimensional Assignment Model. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 1732–1737. [CrossRef]
3. Roy, P.K.; Chowdhary, S.S.; Bhatia, R. A Machine Learning approach for automation of Resume Recommendation system. *Procedia Comput. Sci.* **2020**, *167*, 2318–2327. [CrossRef]
4. Fong, J.; Ocampo, R.; Gross, D.P.; Tavakoli, M. Intelligent robotics incorporating machine learning algorithms for improving functional capacity evaluation and occupational rehabilitation. *J. Occup. Rehabil.* **2020**, *30*, 362–370. [CrossRef] [PubMed]
5. Kim, H.; Kang, W.S.; Park, H.J.; Lee, J.Y.; Park, J.W.; Kim, Y.; Seo, J.W.; Kwak, M.Y.; Kang, B.C.; Yang, C.J.; et al. Cochlear implantation in postlingually deaf adults is time-sensitive towards positive outcome: Prediction using advanced machine learning techniques. *Sci. Rep.* **2018**, *8*, 18004. [CrossRef]
6. Aish, M.A.; Abu-Naser, S.S.; Abu-Jamie, T.N. Classification of Pepper Using Deep Learning. *IJAER*, **2022**, *6*, 24–31.
7. Hassanin, M.; Radwan, I.; Khan, S.; Tahtali, M. Learning discriminative representations for multi-label image recognition. *J. Vis. Commun. Image Represent.* **2022**, *83*, 103448. [CrossRef]
8. Yun, D.; Choi, S.H. Deep Learning-Based Estimation of Reverberant Environment for Audio Data Augmentation. *Sensors* **2022**, *22*, 592. [CrossRef]
9. Boddapati, V.; Petef, A.; Rasmusson, J.; Lundberg, L. Classifying environmental sounds using image recognition networks. *Procedia Comput. Sci.* **2017**, *112*, 2048–2056. [CrossRef]
10. Ling, X.; Dai, W.; Xue, G.R.; Yang, Q.; Yu, Y. Spectral domain-transfer learning. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 24–27 August 2008; pp. 488–496.

11. Ibrahim, A.K.; Zhuang, H.; Cherubin, L.M.; Schärer-Umpierre, M.T.; Nemeth, R.S.; Erdol, N.; Ali, A.M. Transfer learning for efficient classification of grouper sound. *J. Acoust. Soc. Am.* **2020**, *148*, EL260–EL266. [CrossRef]
12. Xiao, Y.; Xing, C.; Zhang, T.; Zhao, Z. An intrusion detection model based on feature reduction and convolutional neural networks. *IEEE Access* **2019**, *7*, 42210–42219. [CrossRef]
13. Bhatnagar, S.; Afshar, Y.; Pan, S.; Duraisamy, K.; Kaushik, S. Prediction of aerodynamic flow fields using convolutional neural networks. *Comput. Mech.* **2019**, *64*, 525–545. [CrossRef]
14. Dong, S.; Quan, Y.; Feng, W.; Dauphin, G.; Gao, L.; Xing, M. A pixel cluster CNN and spectral-spatial fusion algorithm for hyperspectral image classification with small-size training samples. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 4101–4114. [CrossRef]
15. Saeed, N.; Nyberg, R.G.; Alam, M.; Dougherty, M.; Jooma, D.; Rebreyend, P. Classification of the Acoustics of Loose Gravel. *Sensors* **2021**, *21*, 4944. [CrossRef]
16. Zhang, B.; Leitner, J.; Thornton, S. *Audio Recognition Using MEL Spectrograms and Convolution Neural Networks*; Noiselab University of California: San Diego, CA, USA, 2019.
17. Yamashita, R.; Nishio, M.; Do, R.K.G.; Togashi, K. Convolutional neural networks: An overview and application in radiology. *Insights Imaging* **2018**, *9*, 611–629. [CrossRef]
18. Salamon, J.; Jacoby, C.; Bello, J.P. A Dataset and Taxonomy for Urban Sound Research. In Proceedings of the 22nd ACM International Conference on Multimedia (ACM-MM'14), Orlando, FL, USA, 3–7 November 2014; pp. 1041–1044.
19. Wang, J.C.; Wang, J.F.; He, K.W.; Hsu, C.S. Environmental sound classification using hybrid SVM/KNN classifier and MPEG-7 audio low-level descriptor. In Proceedings of the 2006 IEEE International Joint Conference on Neural Network Proceedings, Vancouver, BC, Canada, 16–21 July 2006; pp. 1731–1735.
20. Saki, F.; Kehtarnavaz, N. Background noise classification using random forest tree classifier for cochlear implant applications. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 3591–3595.
21. Zhang, Y.; LV, D.J. Selected features for classifying environmental audio data with random forest. *Open Autom. Control Syst. J.* **2015**, *7*, 135–142. [CrossRef]
22. Pepino, L.; Riera, P.; Gauder, L.; Gravano, A.; Ferrer, L. Detecting distrust towards the skills of a virtual assistant using speech. *arXiv* **2020**, arXiv:2007.15711.
23. Chandio, A.; Shen, Y.; Bendechache, M.; Inayat, I.; Kumar, T. AUDD: Audio Urdu digits dataset for automatic audio Urdu digit recognition. *Appl. Sci.* **2021**, *11*, 8842. [CrossRef]
24. Cui, C.; Ren, Y.; Liu, J.; Chen, F.; Huang, R.; Lei, M.; Zhao, Z. EMOVIE: A Mandarin Emotion Speech Dataset with a Simple Emotional Text-to-Speech Model. *arXiv* **2021**, arXiv:2106.09317.
25. Cowling, M.; Sitte, R. Comparison of techniques for environmental sound recognition. *Pattern Recognit. Lett.* **2003**, *24*, 2895–2907. [CrossRef]
26. Lu, L.; Zhang, H.J.; Li, S.Z. Content-based audio classification and segmentation by using support vector machines. *Multimed. Syst.* **2003**, *8*, 482–492. [CrossRef]
27. Pillos, A.; Alghamidi, K.; Alzamel, N.; Pavlov, V.; Machanavajhala, S. A real-time environmental sound recognition system for the Android OS. In Proceedings of the Detection and Classification of Acoustic Scenes and Events, Budapest, Hungary, 3 September 2016; Department of Signal Processing, Tampere University of Technology: Hervanta, Finland, 2016.
28. Agrawal, D.M.; Sailor, H.B.; Soni, M.H.; Patil, H.A. Novel TEO-based Gammatone features for environmental sound classification. In Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017; pp. 1809–1813.
29. Uzkent, B.; Barkana, B.D.; Cevikalp, H. Non-speech environmental sound classification using SVMs with a new set of features. *Int. J. Innov. Comput. Inf. Control* **2012**, *8*, 3511–3524.
30. Purwins, H.; Li, B.; Virtanen, T.; Schlüter, J.; Chang, S.Y.; Sainath, T. Deep learning for audio signal processing. *IEEE J. Sel. Top. Signal Process.* **2019**, *13*, 206–219. [CrossRef]
31. Li, J.; Dai, W.; Metze, F.; Qu, S.; Das, S. A comparison of deep learning methods for environmental sound detection. In Proceedings of the 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 126–130.
32. Piczak, K.J. Environmental sound classification with convolutional neural networks. In Proceedings of the 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), Boston, MA, USA, 17–20 September 2015; pp. 1–6.
33. Salamon, J.; Bello, J.P. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process. Lett.* **2017**, *24*, 279–283. [CrossRef]
34. Zhou, H.; Song, Y.; Shu, H. Using deep convolutional neural network to classify urban sounds. In Proceedings of the TENCON 2017–2017 IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017; pp. 3089–3092.
35. Zhang, X.; Zou, Y.; Shi, W. Dilated convolution neural network with LeakyReLU for environmental sound classification. In Proceedings of the 2017 22nd International Conference on Digital Signal Processing (DSP), London, UK, 23–25 August 2017; pp. 1–5.
36. Li, S.; Yao, Y.; Hu, J.; Liu, G.; Yao, X.; Hu, J. An ensemble stacked convolutional neural network model for environmental event sound recognition. *Appl. Sci.* **2018**, *8*, 1152. [CrossRef]

37. Copiaco, A.; Ritz, C.; Fasciani, S.; Abdulaziz, N. Scalogram neural network activations with machine learning for domestic multi-channel audio classification. In Proceedings of the 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Ajman, United Arab Emirates, 10–12 December 2019; pp. 1–6.
38. Zeng, Y.; Mao, H.; Peng, D.; Yi, Z. Spectrogram based multi-task audio classification. *Multimed. Tools Appl.* **2019**, *78*, 3705–3722. [CrossRef]
39. Demir, F.; Turkoglu, M.; Aslan, M.; Sengur, A. A new pyramidal concatenated CNN approach for environmental sound classification. *Appl. Acoust.* **2020**, *170*, 107520. [CrossRef]
40. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
41. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
42. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
43. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
44. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
46. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 630–645.
47. Baldassarre, F.; Morín, D.G.; Rodés-Guirao, L. Deep Koalarization: Image Colorization using CNNs and Inception-ResNet-v2. *arXiv* **2017**, arXiv:1712.03400.
48. Da Nóbrega, R.V.M.; Peixoto, S.A.; da Silva, S.P.P.; Rebouças Filho, P.P. Lung nodule classification via deep transfer learning in CT lung images. In Proceedings of the 2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS), Karlstad, Sweden, 18–21 June 2018; pp. 244–249.
49. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2020**, *109*, 43–76. [CrossRef]
50. Raghu, M.; Zhang, C.; Kleinberg, J.; Bengio, S. Transfusion: Understanding transfer learning for medical imaging. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. https://doi.org/10.48550/arXiv.1902.07208
51. Do, C.B.; Ng, A.Y. Transfer learning for text classification. In *Advances in Neural Information Processing Systems 18 (NIPS 2005)*; MIT Press: Cambridge, MA, USA, 2005.
52. Cook, D.; Feuz, K.D.; Krishnan, N.C. Transfer learning for activity recognition: A survey. *Knowl. Inf. Syst.* **2013**, *36*, 537–556. [CrossRef]
53. Han, D.; Liu, Q.; Fan, W. A new image classification method using CNN transfer learning and web data augmentation. *Expert Syst. Appl.* **2018**, *95*, 43–56. [CrossRef]
54. Zaccone, G.; Karim, M.R. *Deep Learning with tensorFlow: Explore Neural Networks and Build Intelligent Systems with Python*; Packt Publishing Ltd.: Birmingham, UK, 2018.
55. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
56. Sandler, M.; Howard, A.G.; Zhu, M.; Zhmoginov, A.; Chen, L. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation. *arXiv* **2018**, arXiv:1801.04381.
57. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *arXiv* **2016**, arXiv:1602.07261.
58. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8697–8710.
59. Ruder, S. An overview of gradient descent optimization algorithms. *arXiv* **2016**, arXiv:1609.04747.
60. Yu, T.; Zhu, H. Hyper-parameter optimization: A review of algorithms and applications. *arXiv* **2020**, arXiv:2003.05689.
61. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
62. Hussain, Z.; Gimenez, F.; Yi, D.; Rubin, D. Differential data augmentation techniques for medical imaging classification tasks. In Proceedings of the AMIA Annual Symposium Proceedings, Washington, DC, USA, 6–8 November 2017; Volume 2017, p. 979.
63. Huang, L.; Pan, W.; Zhang, Y.; Qian, L.; Gao, N.; Wu, Y. Data augmentation for deep learning-based radio modulation classification. *IEEE Access* **2019**, *8*, 1498–1506. [CrossRef]
64. Ornek, A.H.; Ceylan, M. Comparison of traditional transformations for data augmentation in deep learning of medical thermography. In Proceedings of the 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 1–3 July 2019; pp. 191–194.
65. Shijie, J.; Ping, W.; Peiyi, J.; Siping, H. Research on data augmentation for image classification based on convolution neural networks. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 4165–4170.
66. Oppenheim, A.V. Speech spectrograms using the fast Fourier transform. *IEEE Spectr.* **1970**, *7*, 57–62. [CrossRef]

67. Li, J.; Han, L.; Li, X.; Zhu, J.; Yuan, B.; Gou, Z. An evaluation of deep neural network models for music classification using spectrograms. *Multimed. Tools Appl.* **2021**, *8*, 4621–4647. [CrossRef]
68. Dörfler, M.; Bammer, R.; Grill, T. Inside the spectrogram: Convolutional Neural Networks in audio processing. In Proceedings of the 2017 International Conference on Sampling Theory and Applications (SampTA), Tallinn, Estonia, 3–7 July 2017; pp. 152–155.
69. Zhang, T.; Feng, G.; Liang, J.; An, T. Acoustic scene classification based on Mel spectrogram decomposition and model merging. *Appl. Acoust.* **2021**, *182*, 108258. [CrossRef]
70. Hossin, M.; Sulaiman, M.N. A review on evaluation metrics for data classification evaluations. *Int. J. Data Min. Knowl. Manag. Process.* **2015**, *5*, 1.
71. Tang, Z.; Luo, L.; Xie, B.; Zhu, Y.; Zhao, R.; Bi, L.; Lu, C. Automatic Sparse Connectivity Learning for Neural Networks. *arXiv* **2022**, arXiv:2201.05020.
72. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
73. Dai, W.; Dai, C.; Qu, S.; Li, J.; Das, S. Very deep convolutional neural networks for raw waveforms. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 421–425.
74. Chen, Y.; Guo, Q.; Liang, X.; Wang, J.; Qian, Y. Environmental sound classification with dilated convolutions. *Appl. Acoust.* **2019**, *148*, 123–132. [CrossRef]
75. Chandrakala, S.; Jayalakshmi, S. Generative model driven representation learning in a hybrid framework for environmental audio scene and sound event recognition. *IEEE Trans. Multimed.* **2019**, *22*, 3–14. [CrossRef]