


Article

Benchmarking Audio Signal Representation Techniques for Classification with Convolutional Neural Networks

Roneel V. Sharan *, Hao Xiong and Shlomo Berkovsky

Australian Institute of Health Innovation, Macquarie University, Sydney, NSW 2109, Australia; hao.xiong@mq.edu.au (H.X.); shlomo.berkovsky@mq.edu.au (S.B.)

* Correspondence: roneel.sharan@mq.edu.au

Abstract: Audio signal classification finds various applications in detecting and monitoring health conditions in healthcare. Convolutional neural networks (CNN) have produced state-of-the-art results in image classification and are being increasingly used in other tasks, including signal classification. However, audio signal classification using CNN presents various challenges. In image classification tasks, raw images of equal dimensions can be used as a direct input to CNN. Raw time-domain signals, on the other hand, can be of varying dimensions. In addition, the temporal signal often has to be transformed to frequency-domain to reveal unique spectral characteristics, therefore requiring signal transformation. In this work, we overview and benchmark various audio signal representation techniques for classification using CNN, including approaches that deal with signals of different lengths and combine multiple representations to improve the classification accuracy. Hence, this work surfaces important empirical evidence that may guide future works deploying CNN for audio signal classification purposes.



Citation: Sharan, R.V.; Xiong, H.; Berkovsky, S. Benchmarking Audio Signal Representation Techniques for Classification with Convolutional Neural Networks. *Sensors* **2021**, *21*, 3434. <https://doi.org/10.3390/s21103434>

Academic Editors:
David Silvera-Tawil, Qing Zhang and Mahnoosh Kholghi

Received: 30 March 2021
Accepted: 11 May 2021
Published: 14 May 2021

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



Copyright: © 2021 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/>).

Keywords: convolutional neural networks; fusion; interpolation; machine learning; spectrogram; time-frequency image

1. Introduction

Sensing technologies find applications in detecting and monitoring health conditions. For example, audio signals, such as speech, can be useful in detecting anxiety [1] and commanding wheelchair movement [2], acoustic event recognition in elderly care [3], and respiratory sounds in detecting respiratory diseases [4].

Convolutional neural network (CNN) is an established image classification technique that has outperformed conventional methods in various applications, such as in handwritten digit recognition [5] and on the ImageNet dataset [6] containing various image categories. Although deep learning methods, and CNN in particular, were originally designed for large datasets, techniques such as data augmentation [7], transfer learning [8], and regularization [9] have allowed their extension to small datasets with encouraging results [10–13].

Due to such advancements, the robustness of CNN, and forgoing the need for complex feature engineering and extraction required by conventional methods [14–16], it was not long before CNN was adopted in audio signal classification tasks, achieving results superior to conventional techniques [17–19]. However, unlike in image classification, where raw images can be used as a direct input to the CNN, audio signal classification using CNN presents several practical challenges.

Firstly, the raw time-domain signals can be of a varying length [20,21]. Secondly, using time-domain signals for classification with CNN has generally failed to surpass the accuracy achieved with frequency-domain analysis [22–25], which required signal transformation. Finally, feature combination, a commonly used technique for improving the classification performance using conventional classification methods, is not as straightforward to apply to CNN.

Audio signal classification finds various applications and there has been a growing interest in audio signal classification using deep learning and CNN. The advancements in CNN techniques have been covered in several papers [26–30]. Furthermore, while numerous signal representation techniques have been proposed for audio signal classification using CNN, there is a lack of literature critically reviewing and evaluating the various signal representation techniques to be used in conjunction with CNN.

The main contribution of this work is to *overview and benchmark several popular audio signal representation techniques for classification using CNN*. In particular, we focus on time-frequency image representations, time-frequency image resizing techniques to deal with signals of varying lengths, and strategies to combine the learning from different signal representations. The benchmarking results bring to the fore interesting findings about the contribution of these signal representations to the CNN classification accuracy. Our work provides valuable insight for machine learning researchers deploying CNN for audio signal classification tasks.

2. Literature Review

CNN was originally conceived as an image classification technique and one of the challenges in classifying audio signals using CNN has been to find an appropriate image-like representation of the signal. Time-frequency representation of the audio signals is a common approach to forming this image-like representation. Another requirement of CNN is that input images are expected to have the same dimension. This presents another challenge in applications, where the signals are of a different duration, such as in isolated acoustic event and word detection.

Various time-frequency image formation techniques have been proposed for CNN. The conventional spectrogram representation, formed using short-time Fourier transform (STFT) [31,32], is still widely used, such as in speech emotion recognition [33] and spoken digit recognition [25].

This approach, however, has disadvantages. While having a large number of points in computing the Fourier transform can adequately reveal the unique frequency characteristics, it increases the computational costs of CNN. A smaller number of points, on the other hand, may not accurately capture the unique frequency characteristics, resulting in poor classification performance.

A widely used alternative is to use a large transform length and then frequency filterbanks to compute the filterbank energies in various frequency subbands. Two such filters are the moving average filter [34] and mel-filter. The spectrogram representation formed using the moving average filter is called the moving average spectrogram or smoothed-spectrogram [35,36].

The mel-filter, as used in computing mel-frequency cepstral coefficients (MFCCs) [37], has frequency bands equally spaced on the mel-scale [38] resembling the way humans perceive sound. Representations using the mel-scale are popular for use with CNN, as seen in the 2016 Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE) [39]. The image-like representations used with mel-scale are MFCCs, which sometimes include the derivatives (delta and delta-delta coefficients [40]) and the log energies or mel-spectrogram.

In addition, gammatone filters, which model the human auditory system, are used for forming the time-frequency representation of audio signals [41,42], called gammatone-spectrogram or cochleagram. Constant- Q transform (CQT) [43] is another technique for frequency transformation of signal and this is used in time-frequency representation of audio signals [44,45].

Furthermore, in isolated word or acoustic event classification, the duration of the signals can vary greatly. For signals of unequal length, dividing the signal into an equal number of frames is a common approach to obtain the same number of units in the time-frequency representation [18,36]. However, dividing the signal into an equal number of frames can result in small frames for short signals, possibly making it difficult to capture

unique frequency characteristics. An alternative can be to divide the signal into frames of a fixed length, but this will result in a different number of frames for different signals. This was a commonly used technique in computing conventional features, such as MFCCs, whereby the final feature vector could be represented using statistical features, such as mean and standard deviation, computed across all frames. To get an equal sized time-frequency representation, techniques such as image resizing [41,44,46], zero-padding [47,48], and cropping [49] have been deployed.

Moreover, feature combination has been a common practice in various audio classification applications. This allows fusing information acquired from different signal processing techniques and potentially achieving an improved classification performance. Most techniques revolved around combining MFCC features with features such as wavelets, temporal and frequency descriptors, time-frequency image descriptors, and matching pursuit [50], using classifiers such as support vector machine (SVM) and Gaussian mixture model (GMM) [51–54].

With CNN specifically, the concept of feature combination can be realized by using multiple signal representations for classification. Different filters capture frequency characteristics at different centre frequencies and bandwidths. As such, it might be possible to improve CNN using various signal representations. A number of strategies have been proposed to combine the learning from multiple representations [24,45,55,56]. Broadly, the methods can be categorized as early-fusion, mid-fusion, and late-fusion [57–60]. These refer to the classification stage at which the information is combined, such as combining the inputs to the CNN in early-fusion, combining the weights of the middle layers of the CNN in mid-fusion and combining the CNN outputs in late-fusion.

3. Audio Signal Representation Techniques

We discuss the implementation techniques for various time-frequency representations for use with CNN, approaches to deal with signals of different lengths, and signal representation fusion techniques. An overview of the common techniques for forming the time-frequency representations is given in Figure 1. The target time-frequency image dimension is $n_x \times n_y$ where n_x denotes the number of time windows along the x – axis and n_y the number of frequency components along the y – axis. The procedure for computing these time-frequency representations is detailed in the following subsections.

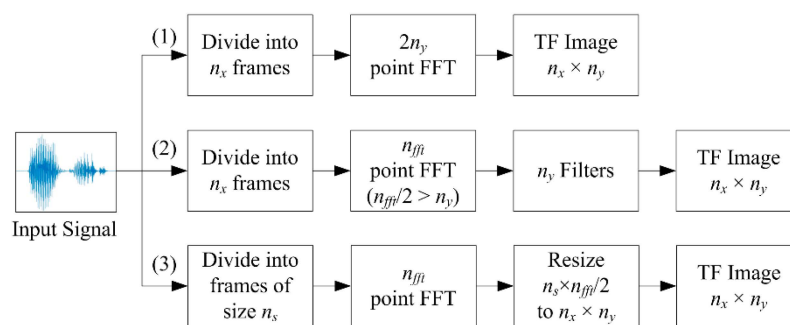


Figure 1. Overview of same sized time-frequency (TF) image formation techniques.

3.1. Time-Frequency Image Representations

In forming the conventional spectrogram (Path (1) in Figure 1), all the signals are divided into n_x frames and a $2n_y$ -point discrete Fourier transform (DFT) is computed. Taking the single-sided spectrum results in a time-frequency representation of size $n_x \times n_y$. The computation of STFT is given as

$$X(k, r) = \sum_{n=0}^{N-1} x(n)w(n)e^{-\frac{2\pi i k n}{N}}, \quad k = 0, \dots, N-1 \quad (1)$$

where N is the length of the window function, $x(n)$ is the time-domain signal, $X(k, r)$ is the k th harmonic for the r th frame, F_s is the sampling frequency, and $w(n)$ is the window function.

The spectrogram values are obtained from log of the DFT values' magnitude as

$$S(k, r) = \log|X(k, r)|^2. \quad (2)$$

In forming the smoothed-spectrogram or mel-spectrogram (Path (2) in Figure 1), the filterbank output of the f th filter is given as

$$E(f, r) = \sum_{k=0}^{\frac{N}{2}-1} V(f, k)|X(k, r)|, \quad f = 1, 2, \dots, F \quad (3)$$

where $V(f, k)$ is the normalized filter response of the moving average filter or mel-filter and F is the total number of filters. The log representation can be computed using (2).

The impulse response of the gammatone filter used for forming the cochleagram representation is given as

$$h(t) = At^{j-1}e^{-2\pi Bt} \cos(2\pi f_c t + \phi) \quad (4)$$

where A is the amplitude, j is the order of the filter, B is the bandwidth of the filter, f_c is the centre frequency of the filter, ϕ is the phase, and t is the time [61].

The gammatone filters are equally spaced on the equivalent rectangular bandwidth (ERB) scale [61]. The three commonly used ERB filter models are given by Glasberg and Moore [62], Lyon's cochlea model [63], and Greenwood [64]. Implementation of a fourth order gammatone filter with four filter stages and each stage being a second order digital filter is described in [65] and a MATLAB implementation is provided in [66].

3.2. Time-Frequency Image Resizing Techniques

Image scaling or resizing using interpolation is a commonly used technique in digital image processing which can be applied to time-frequency image as well (Path (3) in Figure 1). This can be achieved by convolving an image with a small kernel, such as nearest-neighbor, bilinear, bicubic, Lanczos-2, and Lanczos-3 [41,46,67,68].

Nearest neighbour interpolation selects the value of the nearest neighbouring point,

$$R_{NN}(x, y) = S_{[x][y]} \quad (5)$$

the kernel for which in one dimension is given in [69] as

$$k(x) = \begin{cases} 1, & |x| < 0.5 \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Bilinear interpolation, an extension of a linear interpolation in the x and y directions, is given as

$$R_{BL}(x, y) = a_0 + a_1x + a_2y + a_3xy \quad (7)$$

where the coefficients are determined from the four nearest neighbours of (x, y) and implemented using a triangular kernel as

$$k(x) = \begin{cases} 1 - |x|, & |x| < 1 \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

Bicubic interpolation resamples 16 neighbouring pixels as

$$R_{BC}(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij}x^i y^j \quad (9)$$

where the coefficients are determined from the sixteen nearest neighbours of (x, y) and apply convolution with the kernel proposed in [70]

$$k(x) = \begin{cases} \frac{3}{2}|x|^3 - \frac{5}{2}|x|^2 + 1, & |x| \leq 1 \\ -\frac{1}{2}|x|^3 + \frac{5}{2}|x|^2 - 4|x| + 2, & 1 < |x| \leq 2 \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

The Lanczos kernel is a normalized sinc function [71] windowed by the sinc window, which can be equivalently written as

$$L(x) = \begin{cases} 1, & x = 0 \\ \frac{a \sin(\pi x) \sin(\pi x/a)}{\pi^2 x^2}, & -a \leq x < a \text{ and } x \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

where a is a positive integer; the kernel is referred as Lanczos-2 when $a = 2$ and Lanczos-3 when $a = 3$ [72].

The Lanczos interpolation is computed as

$$R_L(x) = \sum_{i=\lfloor x \rfloor - a + 1}^{\lfloor x \rfloor + a} S_i L(x - i) \quad (12)$$

where $\lfloor x \rfloor$ is the floor function of x , a is the filter size, and S_i is a one dimensional signal [69]. The Lanczos kernel in two dimensions is given as $L(x, y) = L(x)L(y)$.

3.3. Combination of Signal Representations

Three common techniques for fusion of time-frequency image representations—early-fusion, mid-fusion, and late-fusion—are illustrated in Figure 2. According to the early-fusion method (Path (1) in Figure 2), multiple representations of the signal are treated as individual channels, similar to a coloured image, on which a single CNN is trained. This technique could also be referred as a multichannel CNN. According to the mid-fusion method (Path (2) in Figure 2), a CNN is trained on each representation of the signal. The activations of all the CNNs are combined and trained in the final layers of the CNN, or in another classifier, to make the final prediction. In late-fusion (Path (3) in Figure 2), CNN outputs trained on the individual representations are fused, e.g., averaging the output score values. The latter two methods could be called multi-input CNN.

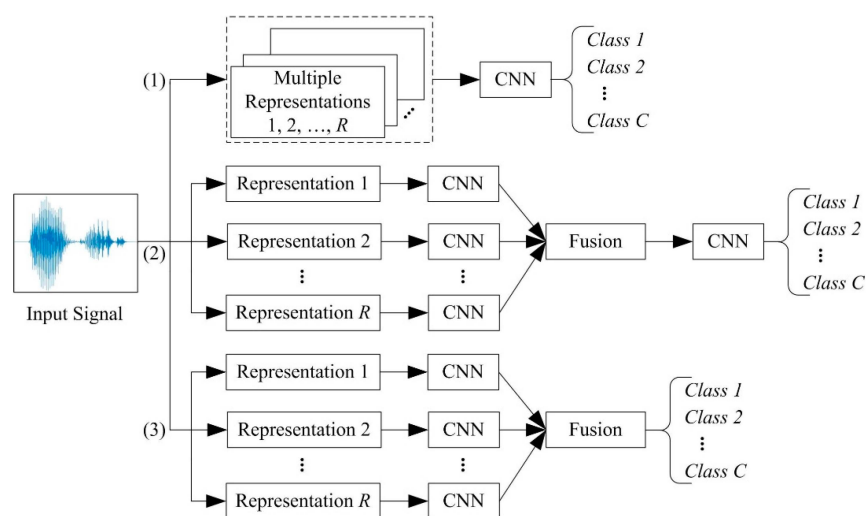


Figure 2. Overview of fusion techniques for learning from multiple representations of the same signal: (1) early-fusion, (2) mid-fusion, and (3) late-fusion.

4. Benchmarking

We evaluate the classification performance of different time-frequency image representation techniques, time-frequency image resizing techniques, and signal representation combination techniques on two audio classification tasks: sound event recognition and isolated word recognition.

4.1. Experimental Setup

4.1.1. Datasets

For the sound event recognition task, we use the Real World Computing Partnership Sound Scene Database (SSD) [73]. The subset of the dataset used in this work has 4000 sound event files, 80 files for each of the 50 classes. The signals are sampled at 44.1 kHz with a 16-bit resolution. The average duration of the segmented signals is 0.5905 s. Furthermore, 50 files from each class are used for training and validating the CNN model in five-fold stratified cross-validation and the remaining 30 are used for testing.

For the isolated word recognition task, we use the Speech Commands dataset [74] which is sampled at 16 kHz. All 105,829 utterances across 35 classes, which have duration of 1 s or less, were used together with 4063 samples of an additional *noise* class generated from six long background noise segments. The dataset was divided into training, validation, and test sets as per the dataset annotation. The noise class was randomly split into the training, validation, and test datasets using the 80%-10%-10% ratio. The final dataset is comprised of 88,093 training segments, 10,387 validation segments, and 11,411 test segments.

In the experiments we report the classification accuracy obtained for the validation and test sets. This communicates the ratio between the number of correctly classified sound events or speech commands and the overall number of classifications.

4.1.2. CNN

The CNN architecture and hyperparameter settings for the two datasets are given in Tables 1 and 2, respectively. A target time-frequency image representation of 32×15 (*height* \times *width*) is used for the sound event dataset. The CNN architecture deployed is similar to that of [36,41] except for the optimization that is performed using adaptive moment estimation (Adam) [75], which was shown to outperform stochastic gradient descent with momentum [76]. The network has two convolutional layers, each of which is followed by batch normalization [77], rectified linear unit (ReLU) [78], and max pooling [79]. This is followed by a fully connected layer and a softmax layer [76].

Table 1. CNN architecture used for the sound event and speech command datasets.

	Sound Event	Speech Command
Image input layer	32×15	64×64
Middle layers	Conv. 1: $16@3 \times 3$, Stride 1×1 , Pad 1×1 Batch Normalization, ReLU Max Pool: 2×2 , Stride 1×1 , Pad 1×1 Conv. 2: $16@3 \times 3$, Stride 1×1 , Pad 1×1 Batch Normalization, ReLU Max Pool: 2×2 , Stride 1×1 , Pad 1×1	Conv. 1: $48@3 \times 3$, Stride 1×1 , Pad 'same' Batch Normalization, ReLU Max Pool: 3×3 , Stride 2×2 , Pad 'same' Conv. 2: $96@3 \times 3$, Stride 1×1 , Pad 'same' Batch Normalization, ReLU Max Pool: 3×3 , Stride 2×2 , Pad 'same' Conv. 3: $192@3 \times 3$, Stride 1×1 , Pad 'same' Batch Normalization, ReLU Max Pool: 3×3 , Stride 2×2 , Pad 'same' Conv. 4: $192@3 \times 3$, Stride 1×1 , Pad 'same' Batch Normalization, ReLU Conv. 5: $192@3 \times 3$, Stride 1×1 , Pad 'same' Batch Normalization, ReLU Max Pool: 3×3 , Stride 2×2 , Pad 'same' Dropout: 0.2
Final layers	Fully connected layer: 50 Softmax layer Classification layer	Fully connected layer: 36 Softmax layer Classification layer

Table 2. Optimization algorithm and hyperparameter settings for training the CNN.

	Sound Event	Speech Command
Optimization algorithm	Adam	Adam
Initial learn rate	0.001	0.0003
Mini batch size	50	128
Max epochs	30	25
Learn rate drop factor	0.5	0.1
Learn rate drop period	6	20
L2 regularization	0.05	0.05

For the speech command dataset, we use a target representation of 64×64 . The CNN architecture is similar to that of [24]. The network has five convolutional layers, each of which is followed by batch normalization and ReLU layers. All ReLU layers, except for the fourth, are followed by a max pooling layer and then the final layers (fully connected and softmax layer).

The early-fusion method is a multichannel CNN, similar to classification of coloured images where the channels represent the R, G, and B image components. For the mid-fusion approach, we found the use of concatenation and addition layers before the fully connected layer to give the best results on the sound event and speech command classification tasks, respectively. The late-fusion approach only requires averaging the probability output of the CNNs trained on the individual representations.

The networks were implemented in MATLAB R2020b and fully trained on AWS using a single NVIDIA V100 Tensor Core GPU.

4.2. Classification Results

4.2.1. Time-Frequency Representations

For the sound event dataset, to form the spectrogram, each signal is divided into 15 frames with a 50% overlap and DFT is computed using 64 points. Computing the single-sided power spectrum results in a 32×15 spectrogram representation. Smoothed-spectrogram and mel-spectrogram use a 1024 point DFT, followed by 32 moving average filters and mel-filters over the single-sided spectrum, respectively, while the cochleagram representation utilises 32 gammatone filters. The speech commands dataset uses a similar approach to form the time-frequency representations. A plot of an example speech command signal *backward* and its four time-frequency representations are shown in Figure 3.

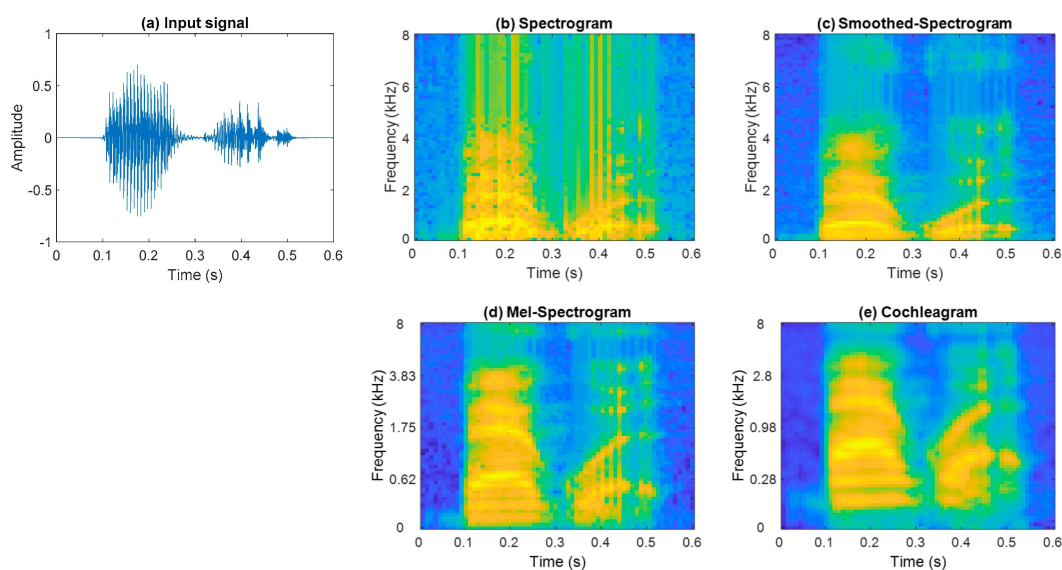


Figure 3. (a) Plot of the speech command signal *backward* and its time-frequency representations: (b) spectrogram, (c) smoothed-spectrogram, (d) mel-spectrogram, and (e) cochleagram.

The classification results in Table 3 show that the use of filterbank energies improves the classification accuracy over the conventional spectrogram. For both of the datasets, the highest classification accuracy is achieved using the cochleagram representation. The relative improvement over the test results using spectrogram representation on the sound event and speech command datasets are 5.16% and 2.43%, respectively. The results suggest that, out of the four time-frequency representations considered for the two audio signal classification tasks, the cochleagram offers the best time-frequency representation for classification using CNN. The finer frequency resolution in the lower frequency range offered by the gammatone filter could explain its robustness in modelling the frequency characteristics of speech and sound event signals [54].

Table 3. Average classification accuracy (in %) of time-frequency image representations.

Signal Representation	Sound Event		Speech Command	
	Validation	Test	Validation	Test
Spectrogram	92.70	93.77	92.33	91.90
Smoothed-Spectrogram	96.48	97.32	93.79	93.41
Mel-Spectrogram	96.45	96.31	93.64	93.64
Cochleagram	98.35	98.61	94.33	94.13

4.2.2. Resized Representations

In this case, the spectrogram representation was formed by dividing each signal into frames of 1024 points for sound events and 256 points for the speech commands. As such, the number of frames was different for signals of different lengths. A 1024 point DFT was then computed and the resulting time-frequency representation was resized to 32×15 for the sound event dataset and to 64×64 for the speech command dataset using interpolation.

The results in Table 4 show that image resizing techniques improve the classification accuracy over the conventional spectrogram representation. The relative improvements in test classification accuracy, with the highest accuracy achieved by the resized spectrogram, are 3.58% and 2.25% on the sound event and speech command datasets, respectively. The best accuracy values are with the bicubic and Lanczos kernel interpolated spectrograms, which could be attributed to their low error in image scaling [41].

Table 4. Average classification accuracy (in %) of resized time-frequency image representations.

Signal Representation	Sound Event		Speech Command	
	Validation	Test	Validation	Test
Resized spectrogram (nearest-neighbour)	93.51	94.19	93.20	93.10
Resized spectrogram (bilinear)	95.71	96.31	94.10	93.81
Resized spectrogram (bicubic)	96.02	96.59	94.03	93.97
Resized spectrogram (Lanczos-2)	95.75	96.42	93.75	93.77
Resized spectrogram (Lanczos-3)	97.01	97.13	94.02	93.75

4.2.3. Fusion Techniques

Classification results using the combined signal representations are given in Table 5. We consider the combination of smoothed-spectrogram, mel-spectrogram, and cochleagram representations. In both datasets, the test results using the signal representation combination techniques are better than the best performing individual cochleagram representation results shown in Table 3. In addition, the classification accuracy using late-fusion is better than mid-fusion and early-fusion. This suggests that while the performance of CNN can be improved using fusion techniques, best results on the two tasks considered

in this work is when CNN is trained independently on each representation and fusion is performed in the end.

Table 5. Average classification accuracy (in %) of signal representation fusion techniques.

Signal Representation Fusion Technique	Sound Event		Speech Command	
	Validation	Test	Validation	Test
Early-fusion	98.42	98.63	94.47	94.29
Mid-fusion	98.48	98.82	94.65	94.49
Late-fusion	98.64	98.83	94.86	94.80

5. Discussion and Conclusions

This paper reviews and evaluates various audio signal representation techniques for classification using CNN. On the sound events and speech commands classification tasks, we reviewed and evaluated the spectrogram, smoothed-spectrogram, mel-spectrogram, and the cochleagram time-frequency representations. While smoothed-spectrogram and mel-spectrogram improved the classification performance over the conventional spectrogram, *the cochleagram representation produced the best classification performance.*

The conventional spectrogram representation offers linearly spaced centre frequencies. On the other hand, the cochleagram representation utilises a gammatone filter, which mimics the human auditory model. The centre frequencies are nonlinearly spaced, having closely spaced centre frequencies in the low frequency range with narrow bandwidth and widely spaced centre frequencies in the upper frequency range. Speech and sound event signals have more frequency content in the lower frequency range, as seen in Figure 3, which is better modelled by the gammatone filter and, thereby, outperforms the conventional methods.

We also considered image resizing techniques, in order to resize time-frequency representations of signals of unequal length. The classification results using the *bicubic and Lanczos kernel interpolations performed best* and were comparable to what could be achieved using smoothed-spectrogram and mel-spectrogram. These interpolation methods offer a lower discrepancy between the interpolated and exact image [41,80], which could explain their better classification performance.

In addition, three techniques for combining multiple signal representations for classification using CNN were reviewed: early-fusion, mid-fusion, and late-fusion. Signal representation combination using the *late-fusion method produced the best classification on both the sound events and speech commands datasets.*

We note that the validation performance on the sound event dataset is, generally, slightly lower than the test performance which could be attributed to the relatively small dataset. During validation, the network was trained on only 40 samples and validated on 10 samples. Once we had settled on the network architecture and tuned the hyperparameters in validation, we trained the network with all 50 training samples (increase of 25% in data over 40 samples) and evaluated the performance on the test data. The increase in training data by 25% could explain the slightly higher performance on the test dataset.

In this work, we limited the evaluation of the audio signal representation combinations to time-frequency representations. However, the studied techniques can be extended to other representations. For example, feature combination of MFCCs and wavelets [52] and MFCCs and time-frequency features [53,54] produced robust classification performance in audio classification tasks. The fusion-based techniques can be extended to these representations as well.

It should be mentioned that the audio signal representation techniques evaluated in this work are for back-end classification using CNN. There is a growing interest in end-to-end CNN models with raw audio signals as input [81–83]. However, a number of these techniques use frequency filters, such as gammatone filters. These are beyond the scope of this work and we plan to study them in the future.

We believe that our findings will be valuable for future works aiming to combine signal processing methods with CNN-based classification tasks. The fact that multiple signal representation methods, datasets, and types of signal were exploited and benchmarked strengthens the validity of our findings and their generalisation potential. Our work surfaces valuable experimental evidence and provides practical guidelines to machine learning researchers, deploying CNN and deep neural networks more generally, to signal classification problems.

Author Contributions: Conceptualization, R.V.S. and S.B.; Methodology, R.V.S. and S.B.; Software, R.V.S. and H.X.; Validation, R.V.S.; Formal Analysis, R.V.S.; Investigation, R.V.S.; Resources, R.V.S. and S.B.; Writing—Original Draft Preparation, R.V.S.; Writing—Review and Editing, R.V.S., S.B. and H.X.; Visualization, R.V.S.; Supervision, S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Salekin, A.; Eberle, J.W.; Glenn, J.J.; Teachman, B.A.; Stankovic, J.A. A weakly supervised learning framework for detecting social anxiety and depression. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **2018**, *2*, 81. [\[CrossRef\]](#)
2. Mazo, M.; Rodríguez, F.J.; Lázaro, J.L.; Ureña, J.; García, J.C.; Santiso, E.; Revenga, P.A. Electronic control of a wheelchair guided by voice commands. *Control. Eng. Pract.* **1995**, *3*, 665–674. [\[CrossRef\]](#)
3. Bonet-Solà, D.; Alsina-Pagès, R.M. A comparative survey of feature extraction and machine learning methods in diverse acoustic environments. *Sensors* **2021**, *21*, 1274. [\[CrossRef\]](#)
4. Sharan, R.V.; Abeyratne, U.R.; Swarnkar, V.R.; Porter, P. Automatic croup diagnosis using cough sound recognition. *IEEE Trans. Biomed. Eng.* **2019**, *66*, 485–495. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Ciregan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3642–3649.
6. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
7. Salamon, J.; Bello, J.P. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process. Lett.* **2017**, *24*, 279–283. [\[CrossRef\]](#)
8. Lisa, T.; Jude, S. Transfer learning. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*; Soria, O.E., Martín, G.J.D., Marcelino, M.-S., Rafael, M.-B.J., Serrano, L.A.J., Eds.; IGI Global: Hershey, PA, USA, 2010; pp. 242–264.
9. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
10. Feng, S.; Zhou, H.; Dong, H. Using deep neural network with small dataset to predict material defects. *Mater. Des.* **2019**, *162*, 300–310. [\[CrossRef\]](#)
11. Liu, S.; Deng, W. Very deep convolutional neural network based image classification using small training sample size. In Proceedings of the 3rd IAPR Asian Conference on Pattern Recognition (ACPR), Kuala Lumpur, Malaysia, 3–6 November 2015; pp. 730–734.
12. Ng, H.-W.; Nguyen, V.D.; Vonikakis, V.; Winkler, S. Deep learning for emotion recognition on small datasets using transfer learning. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, Seattle, WA, USA, 9–13 November 2015; pp. 443–449.
13. Frid-Adar, M.; Diamant, I.; Klang, E.; Amitai, M.; Goldberger, J.; Greenspan, H. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing* **2018**, *321*, 321–331. [\[CrossRef\]](#)
14. Sharan, R.V.; Moir, T.J. An overview of applications and advancements in automatic sound recognition. *Neurocomputing* **2016**, *200*, 22–34. [\[CrossRef\]](#)
15. Gerhard, D. *Audio Signal Classification: History and Current Techniques*; TR-CS 2003-07; University of Regina: Regina, SK, Canada, 2003.
16. Stowell, D.; Giannoulis, D.; Benetos, E.; Lagrange, M.; Plumbley, M.D. Detection and classification of acoustic scenes and events. *IEEE Trans. Multimed.* **2015**, *17*, 1733–1746. [\[CrossRef\]](#)

17. Sainath, T.N.; Mohamed, A.; Kingsbury, B.; Ramabhadran, B. Deep convolutional neural networks for LVCSR. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8614–8618.
18. Abdel-Hamid, O.; Mohamed, A.-R.; Jiang, H.; Deng, L.; Penn, G.; Yu, D. Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 1533–1545. [[CrossRef](#)]
19. Swietojanski, P.; Ghoshal, A.; Renals, S. Convolutional neural networks for distant speech recognition. *IEEE Signal Process. Lett.* **2014**, *21*, 1120–1124. [[CrossRef](#)]
20. Hertel, L.; Phan, H.; Mertins, A. Classifying variable-length audio files with all-convolutional networks and masked global pooling. *arXiv* **2016**, arXiv:1607.02857.
21. Kumar, A.; Raj, B. Deep CNN framework for audio event recognition using weakly labeled web data. *arXiv* **2017**, arXiv:1707.02530.
22. Hertel, L.; Phan, H.; Mertins, A. Comparing time and frequency domain for audio event recognition using deep learning. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Vancouver, CO, Canada, 24–29 July 2016; pp. 3407–3411.
23. Golik, P.; Tüske, Z.; Schlüter, R.; Ney, H. Convolutional neural networks for acoustic modeling of raw time signal in LVCSR. In Proceedings of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH), Dresden, Germany, 6–10 September 2015; pp. 26–30.
24. Sharan, R.V.; Berkovsky, S.; Liu, S. Voice command recognition using biologically inspired time-frequency representation and convolutional neural networks. In Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Montreal, QC, Canada, 20–24 July 2020; pp. 998–1001.
25. Becker, S.; Ackermann, M.; Lapuschkin, S.; Müller, K.-R.; Samek, W. Interpreting and explaining deep neural networks for classification of audio signals. *arXiv* **2018**, arXiv:1807.03418.
26. Ismail Fawaz, H.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P.-A. Deep learning for time series classification: A review. *Data Min. Knowl. Discov.* **2019**, *33*, 917–963. [[CrossRef](#)]
27. Gama, F.; Marques, A.G.; Leus, G.; Ribeiro, A. Convolutional neural network architectures for signals supported on graphs. *IEEE Trans. Signal Process.* **2019**, *67*, 1034–1049. [[CrossRef](#)]
28. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
29. Wang, C.; Tai, T.; Wang, J.; Santoso, A.; Mathulapragansan, S.; Chiang, C.; Wu, C. Sound events recognition and retrieval using multi-convolutional-channel sparse coding convolutional neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 1875–1887. [[CrossRef](#)]
30. Hershey, S.; Chaudhuri, S.; Ellis, D.P.W.; Gemmeke, J.F.; Jansen, A.; Moore, R.C.; Plakal, M.; Platt, D.; Saurous, R.A.; Seybold, B.; et al. CNN architectures for large-scale audio classification. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 131–135.
31. Allen, J. Short term spectral analysis, synthesis, and modification by discrete Fourier transform. *IEEE Trans. Acoust. Speech Signal Process.* **1977**, *25*, 235–238. [[CrossRef](#)]
32. Allen, J. Applications of the short time Fourier transform to speech processing and spectral analysis. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Paris, France, 3–5 May 1982; pp. 1012–1015.
33. Badshah, A.M.; Ahmad, J.; Rahim, N.; Baik, S.W. Speech emotion recognition from spectrograms with deep convolutional neural network. In Proceedings of the International Conference on Platform Technology and Service (PlatCon), Busan, Korea, 13–15 February 2017; pp. 1–5.
34. Brown, R.G. *Smoothing, Forecasting and Prediction of Discrete Time Series*; Dover Publications: Mineola, NY, USA, 2004.
35. Kovács, G.; Tóth, L.; Van Compernelle, D.; Ganapathy, S. Increasing the robustness of CNN acoustic models using autoregressive moving average spectrogram features and channel dropout. *Pattern Recognit. Lett.* **2017**, *100*, 44–50. [[CrossRef](#)]
36. Sharan, R.V.; Moir, T.J. Acoustic event recognition using cochleagram image and convolutional neural networks. *Appl. Acoust.* **2019**, *148*, 62–66. [[CrossRef](#)]
37. Davis, S.; Mermelstein, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech Signal Process.* **1980**, *28*, 357–366. [[CrossRef](#)]
38. Stevens, S.S.; Volkman, J.; Newman, E.B. A scale for the measurement of the psychological magnitude pitch. *J. Acoust. Soc. Am.* **1937**, *8*, 185–190. [[CrossRef](#)]
39. Mesaros, A.; Heittola, T.; Benetos, E.; Foster, P.; Lagrange, M.; Virtanen, T.; Plumbley, M.D. Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *26*, 379–393. [[CrossRef](#)]
40. Young, S.; Evermann, G.; Gales, M.; Hain, T.; Kershaw, D.; Liu, X.; Moore, G.; Odell, J.; Ollason, D.; Povey, D.; et al. *The HTK Book (for HTK Version 3.4)*; Cambridge University Engineering Department: Cambridge, UK, 2009.
41. Sharan, R.V.; Moir, T.J. Time-frequency image resizing using interpolation for acoustic event recognition with convolutional neural networks. In Proceedings of the IEEE International Conference on Signals and Systems (ICSigSys), Bandung, Indonesia, 16–18 July 2019; pp. 8–11.
42. Tjandra, A.; Sakti, S.; Neubig, G.; Toda, T.; Adriani, M.; Nakamura, S. Combination of two-dimensional cochleogram and spectrogram features for deep learning-based ASR. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 4525–4529.

43. Brown, J.C. Calculation of a constant Q spectral transform. *J. Acoust. Soc. Am.* **1991**, *89*, 425–434. [[CrossRef](#)]
44. Rakotomamonjy, A.; Gasso, G. Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 142–153. [[CrossRef](#)]
45. McLoughlin, I.; Xie, Z.; Song, Y.; Phan, H.; Palaniappan, R. Time-frequency feature fusion for noise robust audio event classification. *Circuits Syst. Signal Process.* **2020**, *39*, 1672–1687. [[CrossRef](#)]
46. Ozer, I.; Ozer, Z.; Findik, O. Noise robust sound event classification with convolutional neural network. *Neurocomputing* **2018**, *272*, 505–512. [[CrossRef](#)]
47. Hashemi, M. Enlarging smaller images before inputting into convolutional neural network: Zero-padding vs. interpolation. *J. Big Data* **2019**, *6*, 98. [[CrossRef](#)]
48. Reddy, D.M.; Reddy, N.V.S. Effects of padding on LSTMs and CNNs. *arXiv* **2019**, arXiv:1903.07288.
49. Tang, H.; Ortis, A.; Battiato, S. The impact of padding on image classification by using pre-trained convolutional neural networks. In Proceedings of the 20th International Conference on Image Analysis and Processing (ICIAP), Trento, Italy, 9–13 September 2019; pp. 337–344.
50. Mallat, S.G.; Zhang, Z. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.* **1993**, *41*, 3397–3415. [[CrossRef](#)]
51. Guo, G.; Li, S.Z. Content-based audio classification and retrieval by support vector machines. *IEEE Trans. Neural Netw.* **2003**, *14*, 209–215. [[CrossRef](#)]
52. Rabaoui, A.; Davy, M.; Rossignol, S.; Ellouze, N. Using one-class SVMs and wavelets for audio surveillance. *IEEE Trans. Inf. Forensics Secur.* **2008**, *3*, 763–775. [[CrossRef](#)]
53. Chu, S.; Narayanan, S.; Kuo, C.C.J. Environmental sound recognition with time-frequency audio features. *IEEE Trans. Audio Speech Lang. Process.* **2009**, *17*, 1142–1158. [[CrossRef](#)]
54. Sharan, R.V.; Moir, T.J. Subband time-frequency image texture features for robust audio surveillance. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2605–2615. [[CrossRef](#)]
55. 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](#)]
56. Su, Y.; Zhang, K.; Wang, J.; Madani, K. Environment sound classification using a two-stream CNN based on decision-level fusion. *Sensors* **2019**, *19*, 1733. [[CrossRef](#)]
57. Mesaros, A.; Heittola, T.; Virtanen, T. Acoustic scene classification: An overview of DCASE 2017 Challenge entries. In Proceedings of the 16th International Workshop on Acoustic Signal Enhancement (IWAENC), Tokyo, Japan, 17–20 September 2018; pp. 411–415.
58. Pandeya, Y.R.; Lee, J. Deep learning-based late fusion of multimodal information for emotion classification of music video. *Multimed. Tools Appl.* **2021**, *80*, 2887–2905. [[CrossRef](#)]
59. Wang, H.; Zou, Y.; Chong, D. Acoustic scene classification with spectrogram processing strategies. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020), Tokyo, Japan, 2–4 November 2020; pp. 210–214.
60. Sharan, R.V. Spoken digit recognition using wavelet scalogram and convolutional neural networks. In Proceedings of the IEEE Recent Advances in Intelligent Computational Systems (RAICS), Thiruvananthapuram, India, 3–5 December 2020; pp. 101–105.
61. Patterson, R.D.; Robinson, K.; Holdsworth, J.; McKeown, D.; Zhang, C.; Allerhand, M. Complex sounds and auditory images. In *Auditory Physiology and Perception*; Cazals, Y., Horner, K., Demany, L., Eds.; Pergamon: Oxford, UK, 1992; pp. 429–446.
62. Glasberg, B.R.; Moore, B.C. Derivation of auditory filter shapes from notched-noise data. *Heart Res.* **1990**, *47*, 103–138. [[CrossRef](#)]
63. Slaney, M. *Lyon's Cochlear Model*; Apple Computer: Cupertino, CA, USA, 1988; Volume 13.
64. Greenwood, D.D. A cochlear frequency-position function for several species-29 years later. *J. Acoust. Soc. Am.* **1990**, *87*, 2592–2605. [[CrossRef](#)]
65. Slaney, M. *An Efficient Implementation of the Patterson-Holdsworth Auditory Filter Bank*; Apple Computer, Inc.: Cupertino, CA, USA, 1993; Volume 35.
66. Slaney, M. *Auditory Toolbox for Matlab*; Interval Research Corporation: Palo Alto, CA, USA, 1998; Volume 10.
67. Zhang, W.; Han, J.; Deng, S. Heart sound classification based on scaled spectrogram and partial least squares regression. *Biomed. Signal Process. Control* **2017**, *32*, 20–28. [[CrossRef](#)]
68. Verstraete, D.; Ferrada, A.; Droguett, E.L.; Meruane, V.; Modarres, M. Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings. *Shock Vib.* **2017**, *2017*, 5067651. [[CrossRef](#)]
69. Stallmann, C.F.; Engelbrecht, A.P. Signal modelling for the digital reconstruction of gramophone noise. In Proceedings of the International Conference on E-Business and Telecommunications (ICETE) 2015, Colmar, France, 20–22 July 2016; pp. 411–432.
70. Keys, R. Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process.* **1981**, *29*, 1153–1160. [[CrossRef](#)]
71. Gearhart, W.B.; Shultz, H.S. The function $\sin x/x$. *Coll. Math. J.* **1990**, *21*, 90–99. [[CrossRef](#)]
72. Turkowski, K. Filters for common resampling tasks. In *Graphics Gems*; Glassner, A.S., Ed.; Morgan Kaufmann: San Diego, CA, USA, 1990; pp. 147–165.
73. Nakamura, S.; Hiyane, K.; Asano, F.; Nishiura, T.; Yamada, T. Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition. In Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC 2000), Athens, Greece, 31 May–2 June 2000; pp. 965–968.

74. Warden, P. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv* **2018**, arXiv:1804.03209.
75. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
76. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.
77. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
78. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.
79. Jarrett, K.; Kavukcuoglu, K.; Ranzato, M.A.; LeCun, Y. What is the best multi-stage architecture for object recognition? In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 2146–2153.
80. Getreuer, P. Linear methods for image interpolation. *Image Process. Line* **2011**, *1*, 238–259. [[CrossRef](#)]
81. Kim, T.; Lee, J.; Nam, J. Comparison and analysis of SampleCNN architectures for audio classification. *IEEE J. Sel. Top. Signal Process.* **2019**, *13*, 285–297. [[CrossRef](#)]
82. Sainath, T.N.; Weiss, R.J.; Senior, A.; Wilson, K.W.; Vinyals, O. Learning the speech front-end with raw waveform CLDNNs. In Proceedings of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH), Dresden, Germany, 6–10 September 2015; pp. 1–5.
83. Park, H.; Yoo, C.D. CNN-based learnable gammatone filterbank and equal-loudness normalization for environmental sound classification. *IEEE Signal Process. Lett.* **2020**, *27*, 411–415. [[CrossRef](#)]