applied sciences

MDPI

Article

Sound Event Detection Using Derivative Features in Deep Neural Networks

Jin-Yeol Kwak and Yong-Joo Chung *

Department of Electronics, Keimyung University, Daegu 42601, Korea; kbsong11@naver.com
* Correspondence: yjjung@kmu.ac.kr; Tel.: +82-53-580-5925

Received: 23 June 2020; Accepted: 15 July 2020; Published: 17 July 2020

Abstract: We propose using derivative features for sound event detection based on deep neural networks. As input to the networks, we used log-mel-filterbank and its first and second derivative features for each frame of the audio signal. Two deep neural networks were used to evaluate the effectiveness of these derivative features. Specifically, a convolutional recurrent neural network (CRNN) was constructed by combining a convolutional neural network and a recurrent neural networks (RNN) followed by a feed-forward neural network (FNN) acting as a classification layer. In addition, a mean-teacher model based on an attention CRNN was used. Both models had an average pooling layer at the output so that weakly labeled and unlabeled audio data may be used during model training. Under the various training conditions, depending on the neural network architecture and training set, the use of derivative features resulted in a consistent performance improvement by using the derivative features. Experiments on audio data from the Detection and Classification of Acoustic Scenes and Events 2018 and 2019 challenges indicated that a maximum relative improvement of 16.9% was obtained in terms of the F-score.

Keywords: sound event detection; convolutional recurrent neural network; derivative features; mean-teacher model; attention model

1. Introduction

Humans can obtain information about their surroundings from nearby sounds. Accordingly, sound signal analysis, whereby information may be automatically extracted from audio data, has attracted considerable attention. The Detection and Classification of Acoustic Scenes and Events (DCASE) 2013–2020 challenges have greatly contributed to increasing the interest in this area, and several competition tasks have been defined. Among these tasks, sound event detection (SED) is aimed at identifying both the existence and occurrence times of the various sounds in our daily lives [1]. It has several applications, such as surveillance [2,3], urban sound analysis [4], information retrieval from multimedia content [5], health care monitoring [6], and bird call detection [7].

Recently, deep neural networks (DNNs) have demonstrated superior performance to that of conventional machine learning techniques in image classification [8], speech recognition [9], and machine translation [10]. In [11–13], it was demonstrated that the feedforward neural networks (FNNs) outperformed the traditional Gaussian mixture model and support vector machines in SED. Therefore, current studies on SED primarily focus on DNN-based approaches.

Owing to their fixed interlayer connections, FNNs (which are the basic architecture of DNNs) cannot effectively handle signal distortions in image classification. The same phenomenon may occur in SED, which generally uses a two-dimensional time-frequency spectrogram as input to the FNN. Moreover, FNNs have limitations in modeling the long-term time-correlation of the sound signal samples. Accordingly, FNNs are not widely used in SED.

Convolutional neural networks (CNNs) have exhibited superior performance to that of FNNs in various recognition tasks, including SED [11]. By using two-dimensional filters, the parameters of which are shared along the time and frequency shift, CNNs can efficiently handle audio signal distortions in the time-frequency domain [12]. However, CNNs are not suitable for modeling the time correlations of audio signals.

Recurrent neural networks (RNNs), such as gated recurrent neural networks (GRUs) and long short-term memory, have been successfully used in speech recognition [14]. They can efficiently model the long-term time-correlations in time-series signals, and thus they are expected to facilitate SED.

The recently proposed convolutional recurrent neural networks (CRNNs), which combine CNNs and RNNs, have exhibited satisfactory classification performance in SED [11]. They are currently recognized as a highly effective deep neural network architecture in SED and have been widely used in the DCASE challenge since 2018.

As SED becomes more demanding, larger amounts of training data are required. However, owing to the high cost of gathering strongly labeled data, weakly labeled data are widely used instead. In recent DCASE challenges, weakly labeled and unlabeled data were provided as the training data for CRNNs. Although CRNNs perform quite well in SED, they generally assume strongly labeled training data and are not robust against weakly labeled and unlabeled training data.

Recently, attention-based neural networks have been widely used in speech recognition, vision classification and machine translation [15–18]. In some studies, improved sound classification results have been obtained by using an attention mechanism in the deep neural network. In [19], an attention scheme was incorporated into a deep CRNN, and improved performance was exhibited on a weakly labeled SED task. The attention method was effective in identifying sound events from audio recordings, including noisy sounds.

Owing to their availability, unlabeled data are critical for improved SED. In [20], for efficient use of unlabeled training data, a mean-teacher model based on an attention-based CRNN was proposed for SED and it showed the best performance in DCASE challenge 2018. The architecture and training method of this model is designed to be suitable for unlabeled training data [21]. In the recent DCASE 2019 and 2020 challenges, a CRNN architecture based on the mean-teacher model was chosen as the baseline classification model for SED.

As mentioned previously, deep neural networks for SED have evolved from simple FNNs to recent CRNNs, where an attention-based architecture as well as mean-teacher model-based training and evaluation are used. In the aforementioned studies on SED, the log-mel filterbank was calculated for each frame of the signal and was subsequently used input to the deep neural network. However, it is known that derivative features are frequently used in addition to the static features for improved performances in speech recognition [22]. Considering the characteristic similarity between speech and audio signals, we can expect that SED may be improved by using derivative features. Accordingly, the first and second delta features of the log-mel filterbank were extracted and two-dimensional feature maps were constructed using these features in the time-frequency spectrogram domain. Three feature maps, (corresponding to the first and second delta, and static features) were input to the CRNN for improved SED. The effectiveness of this technique was evaluated using a state-of-the-art CRNN.

The remainder of this paper is organized as follows. In Section 2, we introduce the feature extraction method. The architecture of the CRNN used in this study is explained in Section 3. In Section 4, we introduce the databases and present the experiments. Section 5 concludes the paper.

2. Feature Extraction

2.1. Preprocessing

For the training and testing of the CRNN, the log-mel filterbank (LMFB) was extracted to provide acoustic features to the network, as shown in Figure 1. The audio signal was sampled at 16 kHz, and the short-time Fourier transform (STFT) was computed using a hamming window of length 1024

(64 ms) with an overlap of 360 (41.5 ms) [19]. Sixty-four bands of the mel-scale filterbank outputs from 0 to 16 kHz were obtained using the STFT and then were log-transformed to produce the same dimensional LMFB for each 41.5 ms frame. The feature extraction process generated 240 frames with 64 dimensions for the 10 s clips used for training and testing. After the LMFB was computed, it was normalized by subtracting its mean and dividing by its standard deviation over the entire training data. Subsequently, it was used as input to the CRNN.

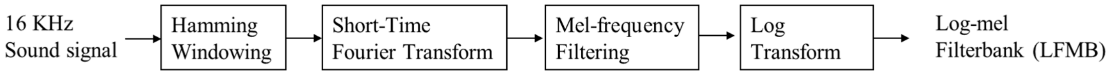

Figure 1. Extraction process of log-mel filterbank (LMFB).

2.2. Derivative Features

In speech recognition, which is similar to SED in that time-series signals are involved, the first and second derivative features are calculated from the static feature. In this study, we consider the LMFB extracted in Figure 1 as the static feature and compute the derivative features of the LMFB to use them for training the CRNN. The computation of the derivative features is done as follows:

$$d_t = \frac{\sum_{k=1}^{K} k(o_{t+k} - o_{t-k})}{2\sum_{k=1}^{K} k^2} \tag{1}$$

where d_t is the derivative feature at time t, and o_t is the static feature. K is the number of frames preceding and following the t-th frame. When computing the second derivative feature, the computed first derivative feature is considered as the static feature in (1).

3. Network Architecture

We used two types of deep neural networks to evaluate the effectiveness of the derivative features in SED: a basic CRNN and a mean teacher model using an attention-based CRNN. As these are considered representative deep neural networks for SED, they may be used to confirm the usefulness of the derivative features.

3.1. Basic CRNN

The architecture of the basic CRNN is shown in Figure 2. It is similar to other CRNNs commonly used for SED [11], but the output of the network is fed to a global average pooling (GAP) layer for training using weakly labeled and unlabeled data. The GAP is used to compute the clip-level output for each class by time-averaging the frame-level sigmoid output of the classification layer. Three convolution blocks (ConvBlocks) consisting of two-dimensional CNNs, one bidirectional GRU layer, and one classification layer implemented as an FNN are cascaded in series. The GAP layer averages the frame-level output of the classification layer for the 240 frames corresponding to the 10 s audio clip. The GPA layer is not used for strongly labeled data.

The 64-dimensional log-mel filterbank and its first and second derivative values are used as input to the basic CRNN. They are independently constructed as (240 × 64)-dimensional feature maps. Although the number of weights increases with the additional feature maps due to the first and second derivative features at the input layer of CRNN, this increase is relatively small compared with the total number of weights of the basic CRNN. The number of weights of the proposed model is about 127,000 compared to 126,000 without derivative features at the input layer.

In ConvBlock, a 3 × 3 convolutional filter is applied to the context window of the input feature map, and batch normalization is used to normalize the filter output to zero mean and unit variance. A rectified linear unit (ReLU) activation function is applied after batch normalization. Non-overlapping 1 × 4 max pooling is applied only in the frequency domain to reduce the dimensionality of the data and to improve frequency invariance. We preserve the time dimension to make use of the time-correlation information of the sound signal, which will be exploited in the following GRU layer. Dropout is used after max pooling to reduce overfitting in the training.

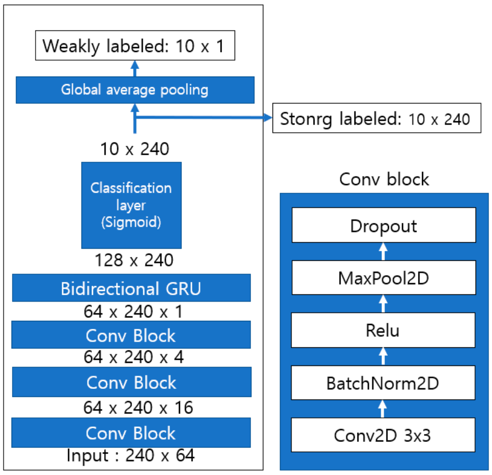

Figure 2. Structure of the basic convolutional recurrent neural network (CRNN).

The output of the last ConvBlock is used as an input to the bidirectional GRU, which has 64 units in each direction and feeds its output into the classification layer, which has 10 units corresponding to the sound classes. These units have a sigmoid activation function, the output of which denotes the posterior probability of the classes for each frame of the sound signal.

3.2. Mean-Teacher Model

The mean-teacher model in this study is similar to that used as the baseline recognizer for SED in the DCASE 2019 challenge [1]. The architecture of the mean-teacher model is shown in Figure 3.

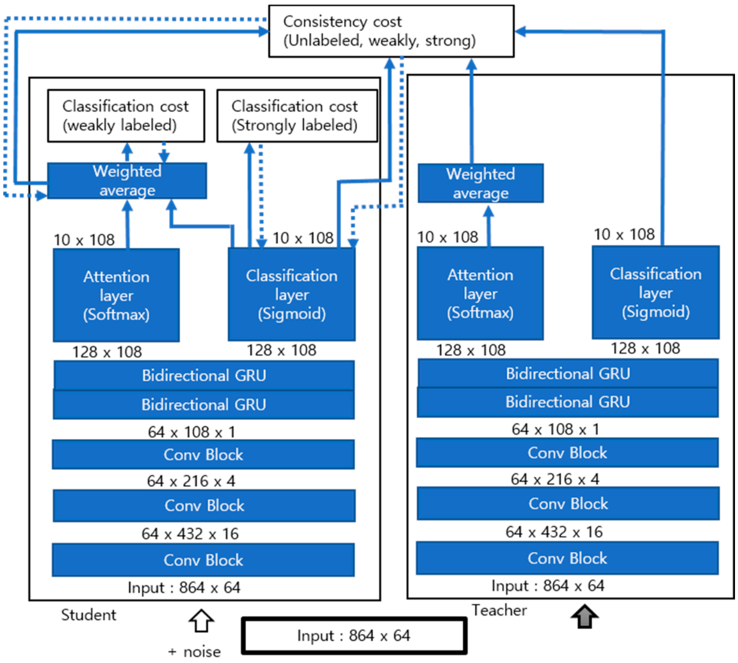

Figure 3. Structure of the mean-teacher model.

It consists of two CRNNs, the student model on the left, and the teacher model on the right. The student model updates the model parameters by calculating the classification and the consistency cost and by back-propagating the errors using gradient descent. The classification cost is calculated by comparing the output of the student model with the ground truth table using the weakly labeled and strongly labeled audio data, as shown in Figure 3. The consistency cost is calculated by comparing the output of the student model with that of the teacher model by using unlabeled, weakly, and strongly labeled data. The teacher model does not update its parameters by back propagation, but uses the ensemble moving average weights of the student model [21]. For the test, the teacher model generally produces more correct output and is used for prediction.

An attention-based CRNN is used for the mean-teacher model. The mechanism is similar to that in [19]. A gated linear unit (GLU) is used in ConvBlock, and an attention layer at the output. The GLU is shown in more detail in Figure 4; it contains a screening module that passes input signals if related to the sound event of interest, and blocks all other the signals.

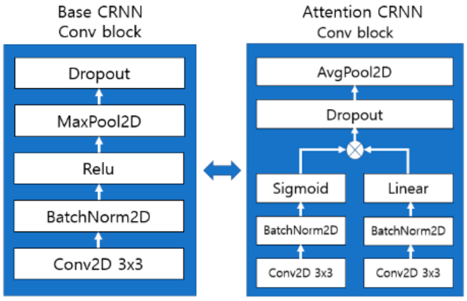

Figure 4. Structure of the gated linear unit (GLU) in the attention CRNN in comparison with the ConvBlock of the basic CRNN.

The attention layer with the softmax activation function is used to compute the weighted average for the output of the classification layer in the 10 s clip. It emphasizes the dominant class at each time step in the computation of the average score of the posterior probability from the classification layer. The weighted average score O_c for the class C is computed by multiplying the attention layer output $O_c^{att}(t)$ and the classification layer output $O_c^{cla}(t)$ as follows.

$$o_c(t) = \frac{\sum_{t=1}^{T} o_c^{att}(t) \times o_c^{cla}(t)}{\sum_{t=1}^{T} o_c^{att}(t)} \quad (2)$$

where T is the total number of time frames in the 10 s audio clip.

4. Experimental Results

4.1. Database

In this study, we used the training and test data of the DCASE 2018 and 2019 challenges. The training set is a combination of the training data from both challenges and consists of weakly labeled, strongly labeled, and unlabeled data. It is presented in Table 1. The weak label provides sound event information at the clip level without timing information at the frame level. Unlabeled data do not have any label information, but we can obtain label information at the clip level by prediction from the CRNN trained using weakly labeled data. The length of each clip is 10 s, and the number of clips for the weakly, strongly and unlabeled data is 1578, 2045 and 14,412, respectively. There are 10 different sound types, usually domestic or household.

Table 1. Training data.

Label Type	Weak Label	Strong Label	Unlabeled
No. of clips	1578	2045	14,412
Properties	Clip-level	Frame-level	None
Clip length	10 s		
Classes (10)	Speech, Dog, Cat, Alarm bell ring, Dishes, Frying, Blender, Running water, Vacuum cleaner, Electric shaver toothbrush		

The details of the testing data are shown in Table 2. The data are divided into the DCASE 2018 and DCASE 2019 test set, which contain 208 and 1168 clips, respectively. The test data contain frame-level label information for the evaluation.

Table 2. Test data.

	DCASE 2018 Test Set	DCASE 2019 Test Set
No. of clips	288	1168
Properties	Frame-level	
Clip length	10 s	
Classes	Same as training data	

4.2. Evaluation Metrics

The CRNN computes the posterior probability for each class in every time frame and identifies a sound event when this probability exceeds 0.5. To improve reliability, a median filtering is applied to the probabilities across the frames before the final decision.

The performance of the CRNN is measured by the F-score and error rate (ER) using an event-based analysis [23], which compares the output of the CRNN with the ground truth table when the output indicates that an event has occurred. The initial decision comprises three different types: true positive (TP), false positive (FP), and false negative (FN). A TP indicates that the period of a detected sound event overlaps with that from the ground truth table. In the decision, a 200 ms onset collar and a 200 ms or 20% of the event length offset collar are allowed. An FP implies that there is no corresponding overlap period in the ground truth table, although the CRNN output indicates an event. An FN implies that there is an event period in the ground truth table, but the CRNN does not produce the corresponding output.

The F-score (F) is computed based on the initial three decisions and is the harmonic average of the precision (P) and recall (R). They are computed as follows.

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \quad \mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \quad \mathrm{F} = \frac{2\mathrm{PR}}{\mathrm{P}+\mathrm{R}} \tag{3}$$

The error rate is computes as

$$\text{Error rate} = \frac{\mathrm{S}+\mathrm{D}+\mathrm{I}}{\mathrm{N}} \tag{4}$$

where N is the total number of sound events active in the ground truth table. Sound events with correct temporal positions but incorrect class labels are counted as substitutions (S), whereas insertions (I) are sound events present in the system output but not in the reference, and deletions (D) are the sound events present in the reference but not in the output [23].

4.3. Experimental Results

To train the basic CRNN, various combinations of weakly and strongly labeled and unlabeled data were considered. Specifically, four combinations, [weakly + unlabeled], [weakly + unlabeled + strongly], [strongly] and [weakly + strongly] were chosen.

Binary cross-entropy was used as the loss function, and the Adam optimizer with a learning rate of 0.001 was used to train the basic CRNN. We applied early stopping with a minimum of five epochs and a patience of 15 epochs. These hyper parameters in this study are based on the baseline systems announced at the DCASE 2018 and 2019 challenges. The classification results on the DCASE 2018 test set are shown in Table 3, where "Single channel" implies that only the static log-mel filterbank was used as the input of the basic CRNN, and "Three channels" implies that derivative features (first and second) were also used as the input.

Table 3. Sound event detection (SED) results for the basic CRNN using derivative features (DCASE 2018 test set).

	DCASE 2018 Test Set			
	Single Channel		**Three Channels**	
	F-Score (%)	**ER**	**F-Score (%)**	**ER**
[weakly + unlabeled] (DCASE 2018 baseline)	12.79 (14.06)	1.44 (1.54)	14.48 -	1.42 -
[weakly + unlabeled + strongly]	17.57	2.42	18.85	2.41
[strongly]	14.99	2.41	16.62	2.51
[weakly + strongly]	15.25	2.42	17.83	2.37
Average	15.15	2.17	16.95	2.18
Average relative improvement	-	-	11.6%	0.5%

It can be seen that [weakly + unlabeled + strongly] yields the best performance because it has the largest amount of training data. However, the performance is not satisfactory considering that the unlabeled training data constitute 80% of all the training data. This implies that the basic CRNN cannot efficiently use the unlabeled data to update its parameters.

Furthermore, using derivative features in all combinations of the training data results in a consistent performance improvement in terms of the F-score. In the [weakly + unlabeled + strongly] combination, which yields the best results, a relative improvement of 7.2% in the F-score is observed. The average relative improvement of all combinations is 11.6%. However, the performance improvement is not sufficiently large to manifest itself in the ER.

We compared the performance of our system with the DCASE 2018 baseline system [24] which uses the same training and test data as the [weakly + unlabeled] combination in Table 3. We could find that, although the baseline system employs similar CRNN architecture as ours, it showed 14.06% in F-score and 1.54 in ER. It has a better F-score and worse ER than the [weakly + unlabeled] combination. Slightly different hyper parameters and differences in the learning process may be the main reasons. However, by using the derivative features, we could obtain better results than the baseline system in both F-score and ER, as shown in the first row of the table.

In Table 4, the classification results using the DCACSE 2019 test set are shown. A similar trend as in Table 3 is shown. The performance improvement by using the derivative features is rather diminished in the DCASE 2019 test set. On average, a 5.3% relative improvement in the F-score is attained, and [weakly + unlabeled + strongly] yields a 6% improvement.

The F-score learning curve during the training of the basic CRNN when only the weakly labeled data are used is shown in Figure 5. Twenty percent of the weakly labeled data were used as validation data. At epoch 45, the optimal performance is attained by early stopping. Further training only increases the F-score of the training data, possibly resulting in overfitting.

Table 4. SED results for the basic CRNN using derivative features (DCASE 2019 test set).

	DCASE 2019 Test Set			
	Single Channel		Three Channels	
	F-Score (%)	ER	F-Score (%)	ER
[weakly + unlabeled]	11.28	1.55	11.93	1.54
[weakly + unlabeled+ strongly]	13.80	2.99	14.63	2.92
[strongly]	12.85	3.07	13.11	3.09
[weakly + strongly]	13.39	2.98	14.41	2.91
Average	12.83	2.65	13.52	2.62
Average relative improvement	-	-	5.3%	1.1%

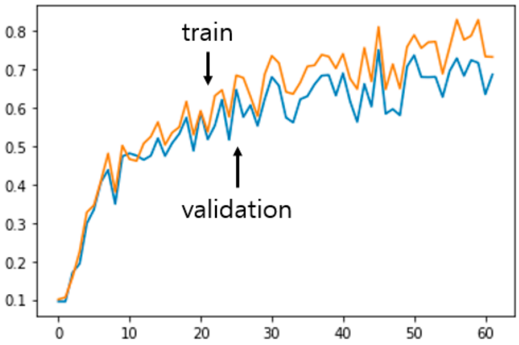

Figure 5. Learning curve in F-score of the basic CRNN with weakly labeled training data.

For the training of the mean-teacher model, we only used the [weakly + unlabeled + strongly] combination because it yielded the best performance in the basic CRNN. Instead of early stopping, the model was trained for 100 and 200 epochs (max epoch = 100 and max epoch = 200 in Tables 5 and 6, respectively), and the best model on the validation data was accordingly selected for the evaluation of the test data. The Adam optimizer was used for training with a learning rate of 0.0001 and a median filtering of length 5 was applied to the output of the classification layer.

The SED results of the mean-teacher model are shown in Table 5. It can be seen that significant performance improvement is obtained by the mean-teacher model (cf. Table 3). Furthermore, the addition of a derivative to the static feature resulted in a 3% relative improvement in the F-score in the DCASE 2018 test set. In the DCASE 2019 test set, a 4.4% relative improvement was attained. In addition, a 2.5% relative improvement was observed when the strongly labeled training data set was used as the test data to demonstrate the effect of the derivative features in case the difference between the testing data and training data is small. It can be concluded that a consistent performance improvement was attained by using the derivative features in the mean-teacher model regardless of the test set: however, the improvement was rather diminished compared with the basic CRNN.

We also compared the performance of our system with the baseline system of the DCASE 2019 [1]. Although the baseline system uses the same audio data and similar mean-teacher model, it showed an F-score of 23.7%, which is worse than our single channel result of 25.95%. As mentioned in the comparison with the baseline of DCASE 2018 in Table 3, the performance difference seems to come from some details in the implementation. However, by using the derivative features, we could further increase the F-score of our system to 27.36%, which is better than the result of the DCASE 2019 baseline system as expected.

The SED results when the training epochs increased to 200 are shown in Table 6. For the DCASE 2018 test set, a 5% relative improvement was attained by using the derivative features. The results for the DCASE 2019 test set indicate a 7.5% improvement. When strongly labeled training data were used as test data, a 2.9% relative improvement was attained. It can be concluded that an increase in the number of training epochs resulted in an increase in the relative improvement by the derivative features.

Table 5. SED results of the mean-teacher model using derivative features (max epoch =100).

	Max Epoch = 100			
	Single Channel		**Three Channels**	
	F-Score (%)	**ER**	**F-Score (%)**	**ER**
DCASE 2018 test set	30.88	1.35	31.82	1.35
Relative Improvement	-	-	3%	0%
DCASE 2019 test set	25.95	1.52	27.09	1.53
(DCASE 2019 baseline)	23.70	-	-	-
Relative Improvement	-	-	4.4%	0%
Strongly labeled training set	68.53	0.58	70.28	0.54
Relative Improvement	-	-	2.5%	6.8%

Table 6. SED results of the mean-teacher model using derivative features (max epoch = 200).

	Max Epoch = 200			
	Single Channel		**Three Channels**	
	F-Score (%)	**ER**	**F-Score (%)**	**ER**
DCASE 2018 test set	31.12	1.32	32.68	1.27
Relative Improvement	-	-	5%	3.8%
DCASE 2019 test set	25.45	1.51	27.36	1.46
Relative Improvement	-	-	7.5%	3.3%
Strongly labeled training set	73.08	0.49	75.17	0.45
Relative Improvement	-	-	2.9%	8.1%

5. Conclusions

Recently, among the approaches for SED, CRNNs have been widely used and have exhibited better performance than other neural networks. In this study, we proposed the use of the first and second delta features of the log-mel filterbank to improve the performance of state-of-the-art CRNNs. We used two types of CRNNs, a basic CRNN and a mean-teacher model based on an attention-based CRNN. We also used various combinations of weakly, strongly labeled, and unlabeled data to train the CRNNs to confirm the effect of the derivative features on SED. Regarding the basic CRNN, a performance improvement was always attained by using the derivative features in the various combinations of the training data. On the DCASE 2018 test set, an 11.6% average relative improvement in the F-score was obtained, and a 5.3% improvement was obtained on the DCASE 2019 test set. Regarding the mean-teacher model, consistent performance improvement was observed as we changed the number of epochs and test set type were changed. When 200 epochs were used for training, a 5% relative improvement in the DCASE 2018 test set and a 7.5% improvement in the DCASE 2019 test set were observed.

In this study, we used the derivative features of the log-mel filterbank to improve SED. In the experiments using various combinations of training and test data, we observed a consistent performance improvement in state-of-the-art CRNNs, which, however, was not as significant as in speech recognition. Nevertheless, the results appear to be sufficient to indicate the importance of the derivative features in SED.

Author Contributions: Conceptualization, Y.-J.C.; methodology, J.-Y.K. and Y.-J.C.; software, J.-Y.K.; validation, J.-Y.K. and Y.-J.C; formal analysis, Y.-J.C.; investigation, Y.-J.C.; data curation, J.-Y.K.; writing—original draft preparation, Y.-J.C.; writing—review and editing, Y.-J.C.; visualization, J.-Y.K.; supervision, Y.-J.C.; project administration, Y.-J.C.; funding acquisition, Y.-J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by Ministry of Education (No. 2018R1A2B6009328).

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

References

1. Turpault, N.; Serizel, R.; Shah, A.; Salamon, J. Sound event detection in domestic environments with weakly labeled data and soundscape synthesis. In Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events, New York, NY, USA, 25–26 October 2019. hal-02160855v2.
2. Nandwana, M.K.; Ziaei, A.; Hansen, J. Robust unsupervised detection of human screams in noisy acoustic environments. In Proceedings of the International Conference on Acoustics, Speech and Signal Processing, Brisbane, Australia, 19–24 April 2015; pp. 161–165.
3. Crocco, M.; Cristani, M.; Trucco, A.; Murino, V. Audio surveillance: A systematic review. *ACM Comput. Surv.* **2016**, *48*, 1–46. [CrossRef]
4. Salamon, J.; Bello, J.P. Feature learning with deep scattering for urban sound analysis. In Proceedings of the 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 31 August–4 September 2015; pp. 724–728.
5. Ntalampiras, S.; Potamitis, I.; Fakotakise, N. On acoustic surveillance of hazardous situations. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Taipei, Taiwan, 19–24 April 2009; pp. 165–168.
6. Wang, Y.; Neves, L.; Metze, F. Audio-based multimedia event detection using deep recurrent neural networks. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 2742–2746.
7. Dekkers, G.; Vuegen, L.; Waterschoot, T.; Vanrumste, B.; Karsmakers, P. DCASE 2018 challenge—Task 5: Monitoring of domestic activities based on multi-channel acoustics. *arXiv* **2018**, arXiv:1807.11246.
8. Krizhevsky, A.; Sutskever, I.; Hinton, G. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Stateline, NV, USA, 3–8 December 2012; pp. 1097–1105.
9. Graves, A.; Mohamed, A.; Hinton, G. Speech recognition with deep recurrent neural Networks. In Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649.
10. Cho, K.; Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1724–1734.
11. Cakir, E.; Parascandolo, G.; Heittola, T.; Huttunen, H.; Virtanen, T. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2017**, *25*, 1291–1303. [CrossRef]
12. Cakir, E.; Heittola, T.; Huttunen, H.; Virtanen, T. Polyphonic sound event detection using multilabel deep neural networks. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–16 July 2015; pp. 1–7.
13. McLoughlin, I.; Zhang, H.; Xie, Z.; Song, Y.; Xiao, W. Robust sound event classification using deep neural networks. *IEEE/ACM Tran. Audio Speech Lang. Process.* **2015**, *23*, 540–552. [CrossRef]
14. Bahdanau, D.; Chorowski, J.; Serdyuk, D.; Brakel, P.; Bengio, Y. End-to-end attention-based large vocabulary speech recognition. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shaghai, China, 20–25 March 2016; pp. 4945–4949.

15. Xu, Y.; Kong, Q.; Huang, Q.; Wang, W.; Plumbley, M.D. Attention and localization based on a deep convolutional recurrent model for weakly supervised audio tagging. In Proceedings of the International Conference on Spoken Language Processing (Interspeech), Stockholm, Sweden, 20–24 August 2017; pp. 3083–3087.
16. Chorowski, K.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. Attention-based models for speech recognition. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 577–585.
17. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent models of visual attention. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2204–2212.
18. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
19. Harb, R.; Pernkopf, F. Sound event detection using weakly labeled semi-supervised data with GCRNNs, VAT and self-adaptive label refinement. *arXiv* **2018**, arXiv:1810.06897.
20. JiaKai, L. Mean teacher convolution system for DCASE 2018 task 4. In Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events, Surrey, UK, 19–20 November 2018.
21. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 1195–1204.
22. Hamid, O.A.; Mohamed, A.; Jiang, H.; Deng, L.; Penn, G.; Yu, D. Convolutional neural networks for speech recognition. *IEEE/ACM Tran. Audio Speech Lang. Process.* **2014**, *22*, 1533–1545. [CrossRef]
23. Mesaros, A.; Heittola, T.; Virtanen, T. Metrics for polyphonic sound event detection. *Appl. Sci.* **2016**, *6*, 162. [CrossRef]
24. Serizel, R.; Turpault, N.; Eghbal-Zadeh, H.; Shah, A.P. Large-Scale Weakly Labeled Semi-Supervised Sound Event Detection in Domestic Environments. In Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events, Surrey, UK, 19–20 November 2018.