Incremental Learning of Acoustic Scenes and Sound Events

Manjunath Mulimani, Annamaria Mesaros

Computing Sciences, Tampere University, Tampere, Finland
manjunath.mulimani@tuni.fi, annamaria.mesaros@tuni.fi

Abstract

In this paper, we propose a method for incremental learning of two distinct tasks over time: acoustic scene classification (ASC) and audio tagging (AT). We use a simple convolutional neural network (CNN) model as an incremental learner to solve the tasks. Generally, incremental learning methods catastrophically forget the previous task when sequentially trained on a new task. To alleviate this problem, we use independent learning and knowledge distillation (KD) between the timesteps in learning. Experiments are performed on TUT 2016/2017 dataset, containing 4 acoustic scene classes and 25 sound event classes. The proposed incremental learner solves the AT task with an F1 score of 54.4% and the ASC task with an accuracy of 88.9% in an incremental time step, outperforming a multi-task system which solves ASC and AT at the same time. The ASC task performance degrades only by 5.1% from the initial time ASC accuracy of 94.0%.

Index Terms: Incremental learning, independent learning, knowledge distillation, acoustic scene classification, audio tagging

1. Introduction

The natural learning system of humans incrementally learns new concepts over time without forgetting the previously learned ones. This process of learning is known as incremental learning. In contrast, deep learning-based systems have the ability to learn a task effectively, but fine-tuning the same system with a new task tends to override the previously acquired knowledge. This leads to a phenomenon of deteriorating performance on previously learned tasks known as catastrophic forgetting. Developing a robust system that should not degrade its performance significantly on previous tasks as new tasks are added is a challenging problem.

Most of the studies reported in the literature on incremental learning operate on images, e.g. object detection [1, 2], image classification [3, 4], and semantic segmentation [5, 6]. For audio classification with real-life audio recordings, sounds are rarely heard in isolation, therefore learning of incremental classes with isolated sound examples is hardly possible. A few works report on incremental learning of audio such as environmental sound classification (ESC) [7, 8], audio captioning [9], and fake audio detection [10]. However, these methods are restricted to solving an initial base task followed by N incremental tasks of a particular problem (e.g. ESC). In addition, these methods require all incremental tasks to have the same number of classes, an assumption that does not hold in real-time scenarios. Furthermore, most of these methods use a small portion of data from the previous task during the training of the system on current task data and complex postprocessing methods to alleviate catastrophic forgetting.

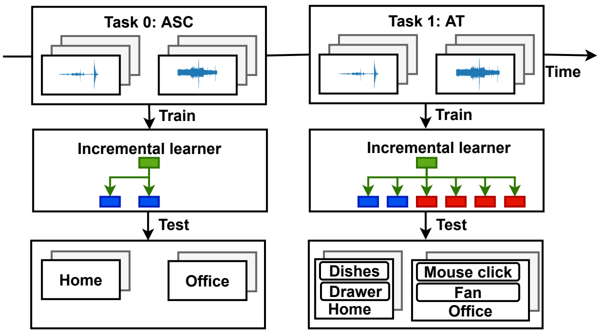

Figure 1: *Incremental learning of distinct tasks: acoustic scene classification (ASC) and audio tagging (AT). Our incremental learner learns acoustic scene classes initially (blue units) and sound event classes incrementally (red units). After the learning process of each task, a learner is evaluated on classes of all tasks learned so far.*

In this work, we propose an incremental learning method to solve distinct tasks over time: acoustic scene classification (ASC) and audio tagging (AT) (see Figure 1), to simulate the scenario of new information becoming available at a later time for the same audio material. In this case the new information is a more detailed characterization of the acoustic content of the scene, i.e. the sound events active in the given acoustic scene. When learning the sound event classes, the same audio material is used, but the acoustic scene labels are no longer available.

A convolutional neural network (CNN) is used as an incremental learner that learns to solve an initial ASC task, and an incremental AT task. The performance of the learner is evaluated on all the tasks learned so far. Unlike existing methods, the proposed incremental learner can deal with the imbalanced classes of distinct tasks. Catastrophic forgetting is reduced using four different approaches: independent learning (IndL) of each task, knowledge distillation (KD), cosine normalized output layer, and reduced learning rate to learn incremental tasks. We also conduct experiments on a three-step incremental learning setup: initial ASC, incremental ASC, and incremental AT tasks. To the best of our knowledge, this is the first work that solves distinct tasks over time.

The rest of the paper is organized as follows. Section 2 introduces the proposed framework for incremental learning of acoustic scenes and sound events tasks. Section 3 presents the different experiments that compare incremental learning with multi-task learning and separate models for the same tasks. Finally, conclusions and future work are given in Section 4.

2. Incremental learning

In an incremental learning framework, a set of tasks $\mathcal{T}_t = \{\tau_0, \tau_1, \ldots, \tau_t\}$ is presented to a learner sequentially in incremental time steps t. The isolated task $\tau_t = \{(\mathbf{x}_i^{\tau_t}, \mathbf{y}_i^{\tau_t})|1 \leq i \leq m\}$ presented at time step t is composed of input features $\mathbf{x}_i^{\tau_t}$ and corresponding one-hot or multi-hot ground truth label vectors $\mathbf{y}_i^{\tau_t} \in \{0,1\}^C$. C denotes the number of classes in tasks up to and including task τ_t. The tasks have disjoint classes and the learner does not have access to the previous classes while learning new classes. This class imbalance creates a bias in the class-specific weights of the last linear layer, making the learner's predictions biased to focus on the classes in the current task and catastrophically forget the classes of the previous task.

In this work, our goal is to build a learner $\mathcal{P}^{\mathcal{T}_t}$, which can solve all the tasks learned so far. A learner $\mathcal{P}^{\mathcal{T}_t}$ is a deep network and it includes a feature extractor ($\mathcal{F}_\theta^{\mathcal{T}_t}$, parameterized by weights θ) and a fully connected layer ($\mathcal{F}_\phi^{\mathcal{T}_t}$, parameterized by weights ϕ) for classification. Output logits of the network on a given input $\mathbf{x}$ are obtained by $\mathbf{o}(\mathbf{x}) = \mathcal{F}_\phi^{\mathcal{T}_t}(\mathcal{F}_\theta^{\mathcal{T}_t}(\mathbf{x}))$.

Generally, incremental learners solve one initial base task followed by N similar incremental tasks. However, in this work, we solve distinct incremental tasks: one is single-label ASC and another is multi-label AT. We experiment with $N = 1$ and 2. For $N = 1$, a learner solves ASC in an initial step ($\tau_0 =$ ASC) and AT in an incremental time step ($\tau_1 =$ AT) as depicted in Fig. 1 (hereafter referred to as ASC-AT task). For $N = 2$, a learner solves ASC in an initial step ($\tau_0 =$ ASC), ASC, and AT in subsequent incremental time steps ($\tau_1 =$ ASC and $\tau_2 =$ AT, hereafter referred to as ASC-ASC-AT).

2.1. Incremental ASC-AT learning

In the initial time step $t = 0$, a learner $\mathcal{P}^{\tau_0}$ learns $\mathcal{F}_\phi^{\tau_0}$ to detect the acoustic scene classes of a task τ_0. $\mathcal{P}^{\tau_0}$ is trained using cross-entropy (CE) loss computed using softmax σ over logits $\mathbf{o}$ as:

$$\mathcal{L}^{CE} = -\sum_{k=1}^{C} \mathbf{y}_k^{\tau_0} \cdot \log(\sigma(\mathbf{o}_k)) \qquad (1)$$

In the next incremental time step $t = 1$, $\mathcal{F}_\phi^{\tau_1}$ is adapted to learn sound event classes of a task τ_1 by adding new output units without changing the other part of the network. This results in a new learner $\mathcal{P}^{\tau_1}$ whose output logits comprise $\mathbf{o} = \{\mathbf{o}^{old}, \mathbf{o}^{new}\}$. $\mathbf{o}^{old}$ and $\mathbf{o}^{new}$ denote the logits of acoustic scene classes and sound event classes respectively.

A weighted binary cross-entropy (BCE) loss is computed using sigmoid σ over logits of the novel acoustic event classes $\mathbf{o}^{new}$ only:

$$\begin{aligned}\mathcal{L}^{BCE} = -\sum_{k=C_0+1}^{C} & w\mathbf{y}_k^{\tau_1} \cdot \log(\sigma(\mathbf{o}_k^{new})) \\ & + (1 - \mathbf{y}_k^{\tau_1}) \cdot \log(1 - \sigma(\mathbf{o}_k^{new})) \qquad (2)\end{aligned}$$

where C_0 denotes the number of classes up to and excluding task τ_1. The weight w reweighs the minor active sound event classes and alleviates the data imbalance caused by a large number of only occasionally occurring sound events and large numbers of inactive frames.

This independent learning (IndL) of weights ϕ of $\mathcal{F}_\phi^{\tau_1}$ of the novel sound event classes from previous acoustic scene classes reduces the weight bias effectively. To be more specific, IndL allows $\mathcal{P}^{\tau_1}$ to learn from Eq. (2) using only the $\mathbf{o}^{new}$ logits of the sound event classes, without disturbing the $\mathbf{o}^{old}$ logits of the acoustic scene classes. This independence of the two tasks is also evident in the different loss functions used in the tasks (CE vs BCE).

Further, in this work, KD is used as a forgetting constraint which penalizes the change concerning the output of the previous learner using Kullback-Leibler divergence ($\mathcal{D}_{KL}$). Note that distillation loss $\mathcal{L}^{KD}$ is always calculated by softmax σ over logits of acoustic scene classes. The output of the current learner $\mathcal{P}^{\tau_1}$, $\mathbf{v}^{old} = \sigma(\mathbf{o}^{old})$ is compared with the output of the previous learner (frozen) $\hat{\mathcal{P}}^{\tau_0}$, $\hat{\mathbf{v}} = \sigma(\hat{\mathcal{P}}^{\tau_0}(\mathbf{x}))$:

$$\mathcal{L}^{KD} = \mathcal{D}_{KL}(\hat{\mathbf{v}}||\mathbf{v}^{old}) \qquad (3)$$

$\mathcal{D}_{KL}$ helps to preserve the learner's knowledge about the previous task. $\mathcal{P}^{\tau_1}$ is trained using combined loss as:

$$\mathcal{L} = \mathcal{L}^{BCE} + \lambda\mathcal{L}^{KD} \qquad (4)$$

2.2. Incremental ASC-ASC-AT learning

In this case, an additional ASC task τ_1 is included before the AT task. ASC task τ_0 and AT task τ_2 are solved as per the description given in 2.1. A learner $\mathcal{P}^{\tau_1}$ is obtained by adding new output units to the previous learner $\mathcal{P}^{\tau_0}$. $\mathcal{P}^{\tau_1}$ then continues to learn to solve another ASC task in the absence of τ_0 data. Taking advantage of IndL, $\mathcal{L}^{CE}$ is computed independently using softmax σ over logits of the new acoustic scene classes $\mathbf{o}^{new}$ only, as per Eq. 1.

In contrast, a learner $\mathcal{P}^{\tau_1}$ without IndL would learn from $\mathcal{L}^{CE}$ using softmax σ over all logits, i.e. $\mathbf{o}^{old}$ and $\mathbf{o}^{new}$, resulting in a bias toward what is sees now, i.e. τ_1 classes; because the learner sees no τ_0 data $(\mathbf{x}^{\tau_0}, \mathbf{y}^{\tau_0})$, it eventually forgets the previously seen acoustic classes.

Further, $\mathcal{L}^{KD}$ is computed along with IndL to preserve the knowledge from the previous acoustic scene classes as per Eq. 3. $\mathcal{P}^{\tau_1}$ is trained using combined loss as:

$$\mathcal{L} = \mathcal{L}^{CE} + \lambda\mathcal{L}^{KD} \qquad (5)$$

2.3. Reducing weight bias further

Independent learning of current and previous classes does not eliminate the weight bias completely. Hence, two more approaches reported in the literature for incremental learning of images are used to further reduce the weight bias. One is, that the learning rate is reduced in incremental time steps, as done in [3]. This improves the transfer of knowledge from the old to the new learner and mitigates the adverse effect of imbalanced data in incremental time steps. Another is the use of cosine normalization in the $\mathcal{F}_\phi^{\mathcal{T}_t}$ layer [11]. It was observed that the magnitude of the weight and bias of the $\mathcal{F}_\phi^{\mathcal{T}_t}$ is much higher than that of the old classes, which leads to $\mathcal{F}_\phi^{\mathcal{T}_t}$ being biased and generates results in favor of new classes. Cosine normalization restricts the values of input distributions to $[-1, 1]$ and eliminates the bias due to the magnitude difference.

3. Evaluation and Results

3.1. Datasets

For ASC-AT, we use acoustic scenes and corresponding sound events from TUT 2016/2017 dataset [12, 13]. The dataset contains 192 minutes of audio recordings. Task 0 is composed of four acoustic scenes: home, residential area, city center, and office. Task 1 is composed of 25 sound events: bird singing, brakes squeaking, breathing, etc. Complete details about the data can be found in [12]. We use this data in two settings, one with ten-second segments and another with one-second segments.

For ASC-ASC-AT, we use TUT Acoustic Scenes 2017 [14] and TUT 2016/2017. Task 0 is composed of 11 acoustic scenes: beach, bus, cafe/restaurant, car, forest path, grocery store, library, metro station, park, train, and tram. Tasks 1 and 2 are the ASC and AT from the previous experiment. The learner is trained and tested on official development and evaluation splits of the datasets in each step.

3.2. Implementation details

Input features to the learner in each time step are 40-dimensional log mel-band energies obtained from each audio segment in 40 ms frames with 50% overlap extracted using Librosa.

The network architecture of the feature extractor $\mathcal{F}_\theta^{\mathcal{T}_t}$ is inspired by the PANNs CNN architecture [15] and implemented using PyTorch. It includes three convolutional blocks. Each block consists of two 3×3 convolutional layers, with batch-normalization and ReLU nonlinearity applied to each convolutional layer. 2×2 average pooling is applied to each convolutional block, and 20% dropout is applied after each average pooling to avoid overfitting. The number of feature maps of convolutional blocks is set to $\{16, 32, 64\}$. The flattened output of the last convolutional block is considered as the input to the cosine normalized fully-connected layer $\mathcal{F}_\phi^{\mathcal{T}_t}$. The number of output units in $\mathcal{F}_\phi^{\mathcal{T}_t}$ is equal to the number of classes in each time step.

The combined learner network is trained using the SGD optimizer [16] with a momentum of 0.9 and a mini-batch size of 100 for 120 epochs. The initial learning rates for task 0 and incremental task(s) are set to 0.1 and 0.01 respectively. CosineAnnealingLR [16] scheduler is used to update the optimizer in every epoch. Other hyper-parameters: w (given in Eq. (2)) and λ (given in Eq. (4) and (5)) are set adaptively as per [17] and [11] respectively.

3.3. Baseline systems and evaluation metrics

The performance of the proposed incremental ASC-AT and ASC-ASC-AT systems are compared with the individual ASC, AT, and joint ASC-AT baseline systems. The network architecture of the incremental ASC-AT system is used in all the baseline systems for pair comparison.

Individual ASC and AT systems solve ASC and AT tasks respectively. A joint ASC-AT system is a multi-task system that solves both ASC and AT tasks at the same time using cross-entropy loss and weighted binary cross-entropy loss respectively. These two losses are combined as per the recommendation of [18] to train the system.

The performance on ASC and AT tasks is evaluated using accuracy and F1 score (using a threshold of 0.5), respectively. For the experiment using one-second segments, the performance on the AT task is evaluated using the error rate [19] as well, because it is equivalent to the segment-based SED evaluation in one second segments.

3.4. ASC-AT results

Table 1: *Incremental ASC (4 classes) and AT (25 classes) for audio segments of 10s with (w/) and without (w/o) KD, compared to separate and joint learning. The value within () denotes the forgetting amount; ↓ indicates that lower is better.*

Method	Task 0	Task 1	
	ASC (Acc)	AT (F1)	ASC (Acc)
ASC	94.0	-	-
AT	-	53.0	-
Joint ASC-AT	72.0	50.4	-
Incremental ASC-AT w/o KD	94.0	54.4	84.1 (9.9↓)
Incremental ASC-AT w/ KD	94.0	54.4	88.9 (5.1↓)

Table 2: *Incremental ASC (4 classes) and AT (25 classes) for audio segments of 1s, compared to individual tasks*

Method	Task 0	Task 1		
	ASC (Acc)	AT (F1)	AT (ER)	ASC (Acc)
ASC	86.8	-	-	
AT	-	46.3	0.70	-
Incremental ASC-AT w/ KD	86.8	47.1	0.71	82.9 (3.9↓)

The experimental results are provided in Tables 1 and 2, comparing the performance of the proposed incremental ASC-AT system with baseline systems on ten-second and one-second audio segments, respectively. Performance scores of individual ASC and AT systems can be considered as upper bounds for other systems. As seen in Table 1, learning of ASC and AT tasks by a single system at the same time as joint ASC-AT results in an overall performance reduction. Particularly, the accuracy of the ASC side in joint ASC-AT is worse than the accuracy of the individual ASC system. This is also true with existing ASC-sound event detection (SED) multi-task models [18, 13]. On the other hand, the proposed incremental ASC-AT method without KD can solve ASC and AT tasks with an accuracy of 84.1% and an F1 score of 54.4% respectively. The performance on the ASC task at the incremental time step is reduced with an absolute difference of 9.9% (forgetting) as compared to the initial time step. Surprisingly, we see a small increase in the F1 score of the AT task as compared to the individual AT system. This may be due to the incremental learner already pre-trained using the same acoustic content (but different classes) in the initial time step, which may generate richer feature representations for the AT task.

As mentioned earlier, KD helps the learner to effectively discriminate between current and previous tasks by preserving the knowledge of the previous task and reducing weight bias. The incremental ASC-AT system with KD outperforms the system without KD, having an average accuracy of 88.9% on the previous ASC task with only 5.1 p.u. forgetting.

For pair comparison with existing individual SED/joint ASC-SED systems, we evaluate the performance of the proposed incremental ASC-AT system on one-second segments as well. Many studies on SED report the segment-based metrics in one-second segments, therefore our AT task in one-second segments provides a close approximation for coarse SED. The results in Table 2 show that shorter segments provide less infor-

Table 3: *Incremental ASC-ASC-AT w/ and w/o KD and independent learning (IndL) in multiple steps (Task 0: 11 classes, Task 1: 4 classes, Task 2: 25 classes)*

Method	Task 0	Task 1		Task 2		
	ASC (Acc)	ASC (Acc)		AT (F1)	ASC (Acc)	
Incremental ASC-ASC-AT w/o (KD + IndL)	65.3	26.4	Task 0: 0.1 (65.2↓) Task 1: 85.0	53.0	23.2	Task 0: 0.1 (65.2↓) Task 1: 80.4 (4.6↓)
Incremental ASC-ASC-AT w/ (KD + IndL)	65.3	53.8	Task 0: 54.6 (10.7↓) Task 1: 53.1	53.0	52.2	Task 0: 53.3 (12↓) Task 1: 49.1 (4↓)

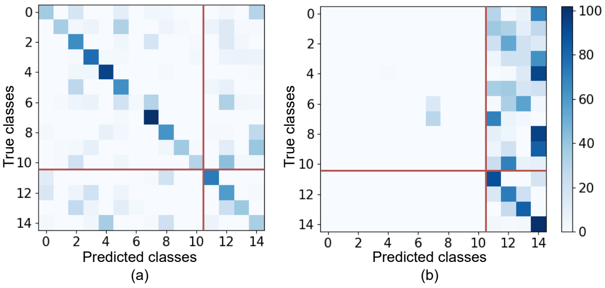

Figure 2: *Confusion matrices w/ and w/o KD and IndL for ASC-ASC incremental learning (Task 0 to Task 1 of Table 3). Red lines separate the regions of new and old classes.*

mation to the learner for decision-making, resulting in lower performance as compared to the results on ten-second segments. Nevertheless, the proposed incremental ASC-AT system achieves an F1 score of 47.1% and an error rate of 0.71 on the AT task, similar to recently reported joint ASC-SED performance on the same dataset [13].

3.5. ASC-ASC-AT results

We also evaluate an ASC-ASC-AT scenario, using ten-second audio segments. The experimental results given in Table 3 demonstrate the effectiveness of independent learning of acoustic scenes (as $\mathbf{o}^{old}, \mathbf{o}^{new}$) from tasks 0 and 1. ASC-ASC tasks have different audio recordings, in contrast to ASC-AT which has the same audio recordings with different class labels.

A learner without IndL learns from Eq. (1) using logits $\mathbf{o}$ (combination of both $\mathbf{o}^{old}, \mathbf{o}^{new}$) and class labels $\mathbf{y}^{\tau_1}$. Because when learning the incremental ASC task 1, the learner does not have access to the data $(\mathbf{x}^{\tau_0}, \mathbf{y}^{\tau_0})$ of previous ASC task 0, the values of the task 0 classes in the $\mathbf{y}^{\tau_1}$ are zeros. This makes the learner forget the old acoustic scene classes, because it sees no examples of them. Hence, the amount of forgetting reached 65.2% and accuracy dropped to almost zero on the old acoustic scene classes at time step 1. In contrast, IndL of current ASC task 1 with KD from previous task does not much disturb the weights of the ASC task 0 and achieve an accuracy of 54.6% with a 10.7% forgetting, without retraining the learner on the old acoustic scene classes. The two cases are illustrated in Figure 2. It can be seen that without KD and IndL the network mostly predicts new classes (Figure 2b), while using KD and IndL rebalances the output (Figure 2a).

It is worth noting that the F1 score of the AT task at $t = 2$ is unaffected and it remains at 53.0%, the same as the individual AT system. This is because the AT task is always learned independently of acoustic scenes and takes advantage of a previous model trained on similar acoustic material, even though the class information differs. This suggests that it is best to use an independent learning mechanism for all the tasks to alleviate catastrophic forgetting effectively.

4. Conclusions and future work

In this paper, we presented incremental ASC-AT and ASC-ASC-AT systems to solve distinct tasks over time. Results show that the performance of the ASC-AT system is close to the individual ASC and AT systems and outperforms the joint ASC-AT learning. Independent learning of previous and current tasks with knowledge distillation significantly reduced the problem of catastrophic forgetting. The learner has also adapted the new classes effectively in incremental time steps. Our method may be considered a good base model for future methods for incremental learning of audio tasks.

In the presented setup, the AT task is always independent of the ASC task. Hence, these two distinct tasks do not much disturb the performances of one another irrespective of their order (whether it is ASC-AT or AT-ASC). In this work, we mainly highlighted the effectiveness of IndL for both distinct and the same types of incremental tasks, using and adapting techniques proposed in literature. Future work also includes more detailed, comprehensive ablation studies of these different choices used to reduce weight bias and improve the overall performance, such as the order and size of the incremental tasks, use of cosine normalization, and use of reduced learning rates in incremental tasks. In addition, we will consider incremental learning of combinations of audio tasks that include the temporal or spatial aspect of the data, such as ASC-SED, SEL-SED, ASC-SELD, etc.

5. Acknowledgements

This work was supported in part by the Academy of Finland grant 332063 “Teaching machines to listen”. The authors wish to thank CSC-IT Centre of Science Ltd., Finland, for providing computational resources.

6. References

[1] J. Kj, J. Rajasegaran, S. Khan, F. S. Khan, and V. N. Balasubramanian, “Incremental object detection via meta-learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[2] J.-M. Perez-Rua, X. Zhu, T. M. Hospedales, and T. Xiang, “Incremental few-shot object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 846–13 855.

[3] S. Mittal, S. Galesso, and T. Brox, “Essentials for class incremental learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3513–3522.

[4] H. Ahn, J. Kwak, S. Lim, H. Bang, H. Kim, and T. Moon, “SS-IL: Separated softmax for incremental learning,” in *Proceedings*

of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 844–853.

[5] L. Yu, X. Liu, and J. Van de Weijer, “Self-training for class-incremental semantic segmentation,” *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[6] F. Cermelli, D. Fontanel, A. Tavera, M. Ciccone, and B. Caputo, “Incremental learning in semantic segmentation from image labels,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4371–4381.

[7] Y. Wang, N. J. Bryan, M. Cartwright, J. P. Bello, and J. Salamon, “Few-shot continual learning for audio classification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 321–325.

[8] B. Bayram and G. İnce, “An incremental class-learning approach with acoustic novelty detection for acoustic event recognition,” *Sensors*, vol. 21, no. 19, p. 6622, 2021.

[9] J. Berg and K. Drosos, “Continual learning for automated audio captioning using the learning without forgetting approach,” in *Detection and Classication of Acoustic Scenes and Events*, 2021, pp. 140–144.

[10] H. Ma, J. Yi, J. Tao, Y. Bai, Z. Tian, and C. Wang, “Continual learning for fake audio detection,” *arXiv preprint arXiv:2104.07286*, 2021.

[11] S. Hou, X. Pan, C. C. Loy, Z. Wang, and D. Lin, “Learning a unified classifier incrementally via rebalancing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 831–839.

[12] K. Imoto, N. Tonami, Y. Koizumi, M. Yasuda, R. Yamanishi, and Y. Yamashita, “Sound event detection by multitask learning of sound events and scenes with soft scene labels,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 621–625.

[13] D. A. Krause and A. Mesaros, “Binaural signal representations for joint sound event detection and acoustic scene classification,” in *European Signal Processing Conference (EUSIPCO)*, 2022, pp. 399–403.

[14] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, “DCASE 2017 challenge setup: Tasks, datasets and baseline system,” in *Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2017.

[15] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “PANNs: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[16] I. Loshchilov and F. Hutter, “SGDR: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.

[17] “PyTorch documentation,” https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html.

[18] N. Tonami, K. Imoto, M. Niitsuma, R. Yamanishi, and Y. Yamashita, “Joint analysis of acoustic events and scenes based on multitask learning,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2019, pp. 338–342.

[19] A. Mesaros, T. Heittola, and T. Virtanen, “Metrics for polyphonic sound event detection,” *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.