



Audio Pyramid Transformer with Domain Adaption for Weakly Supervised Sound Event Detection and Audio Classification

Yifei Xin, Dongchao Yang, Yuexian Zou*

ADSPLAB, School of ECE, Peking University, Shenzhen, China

{xinyifei,dongchao98}@stu.pku.edu.cn, zouyx@pku.edu.cn

Abstract

Recently, the Transformer-based model has been applied to sound event detection and audio classification tasks. However, when processing the audio spectrogram on a fine-grained scale, the computational cost is still high even with a hierarchical structure. In this paper, we introduce APT: an audio pyramid transformer with quadtree attention to reduce the computational complexity from quadratic to linear. Besides, most previous methods for weakly supervised sound event detection (WSSSED) utilize the multi-instance learning (MIL) mechanism. However, MIL focuses more on the accuracy of bags (clips) rather than the instances (frames), so it tends to localize the most distinct part but not the whole sound event. To solve this problem, we provide a novel perspective that models WSSSED as a domain adaption (DA) task, where the weights of the classifier trained on the source (clip) domain are shared to the target (frame) domain to enhance localization performance. Furthermore, we introduce a DAD (domain adaption detection) loss to align the feature distribution between frame and clip domain and make the classifier perceive frame domain information better. Experiments show that our APT achieves new state-of-the-art (SOTA) results on AudioSet, DCASE2017 and Urban-SED datasets. Moreover, our DA-WSSSED pipeline significantly outperforms the MIL-based WSSSED method.

Index Terms: Weakly Supervised Sound Event Detection, Audio Classification, Pyramid Transformer, Domain Adaption

1. Introduction

Transformers can capture long-range dependencies by the attention mechanism [1] and have demonstrated tremendous success in computer vision fields. Recently, the audio spectrogram transformer (AST) [2] which uses the Vision Transformer (ViT) [3] backbone is directly applied to an audio spectrogram for audio classification tasks. Although AST achieves competitive performance, it consumes high computational and memory costs as its columnar structure and full attention mechanism [1] with quadratic complexity. To reduce the computational cost, HTS-AT [4] uses the Swin Transformer [5] backbone to restrict the attention in local windows in a single attention block, which hurts long-range dependencies, the most significant merits of transformers. These shortcomings in audio transformers motivate us to explore more advanced attention mechanisms which can achieve less information loss while keeping high efficiency.

Sound event detection (SED) consists of two subtasks, one is to tag the absence or presence of audio events in an audio clip, and the other is to locate their corresponding onset and offset times. Recently, WSSSED has gained an increasing attention as weak labels are much easier to gather than strong labels. A common framework for WSSSED is multiple instance learning

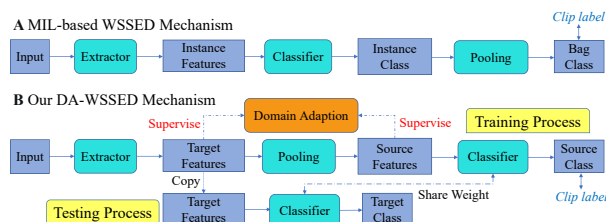


Figure 1: The comparison of MIL-based WSSSED and our DA-WSSSED pipeline.

(MIL) in which the dataset labels are bags of classes. In MIL, the bag label is positive if and only if the bag contains at least one positive instance. As shown in Figure 1 A, the neural network (i.e., the Extractor with Classifier) predicts the probability of each sound event type being active at each frame. Then, a pooling function aggregates the frame-level probabilities into a clip-level probability for each sound event type. However, the MIL-based WSSSED method pays more attention to the accuracy of bags (clips) rather than the instances (frames), thereby having certain limitations in locating the entire sound event.

Domain adaption (DA) aims to learn a discriminative model in the presence of variations between distributions of the training and testing samples [6]. By treating clip-level and frame-level features as the features respectively extracted from source and target domain, the goal of WSSSED is consistent with the DA task, i.e., forcing the classifier trained on the source (clip) domain to perform well on the target (frame) domain. Therefore, if the DA method can help WSSSED align the distribution of these two domains, the classifier can avoid overfitting the source domain, i.e., only recognizing the salient parts of recordings. Inspired by this, we construct a DA-WSSSED pipeline that helps to better engage the DA approaches into WSSSED to enhance the performance. Figure 1 B provides a visual overview of our DA-WSSSED pipeline. Compared with the MIL mechanism, our method employs DA to align the feature distribution between source and target domain, which can improve the accuracy of the classifier when projected to the frame-level features.

In this paper, we introduce APT, an audio pyramid transformer with domain adaption method for WSSSED and audio classification. Our contributions can be listed as:

- APT with quadtree attention achieves new SOTAs on AudioSet, DCASE2017 and Urban-SED datasets. Moreover, APT without ImageNet-pretraining still achieves competitive performance.
- APT reduces the computational complexity from quadratic to linear without compromising performance.
- Our work is the first to model WSSSED as a DA task and design a DA-WSSSED pipeline to assist WSSSED by DA methods. Furthermore, a DAD loss is introduced

*Corresponding author

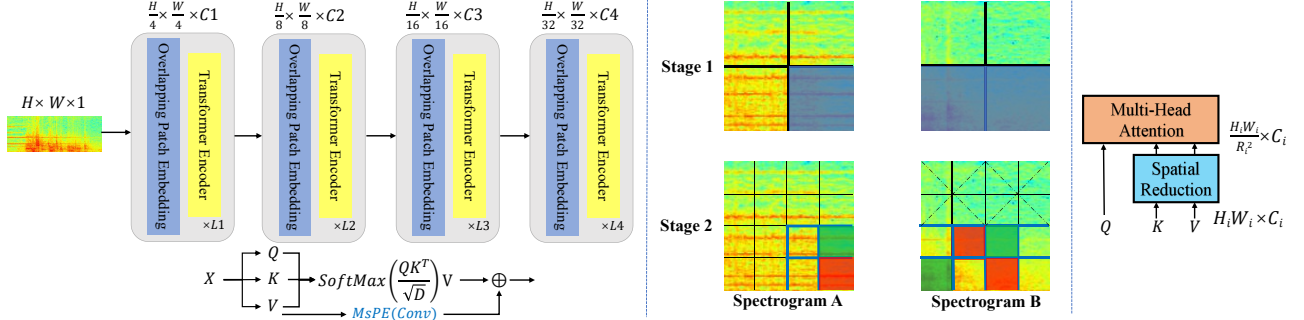


Figure 2: *Left: the overall architecture of APT and the multi-scale position encoding (MsPE), Medium: the illustration of quadtree attention (QA), Right: spatial-reduction attention (SRA).*

to align the feature distribution of source and target domain in the WSSSED scenario. Experiments show that our DA-WSSSED method outperforms the MIL-based WSSSED mechanism on the weakly labeled DCASE2017 and Urban-SED datasets on multiple metrics.

2. Audio Pyramid Transformer

2.1. Overall Architecture

The overall architecture of APT is shown in the left of Figure 2. To produce a hierarchical representation, the whole network consists of four stages. All stages share a similar architecture, which consists of a patch embedding layer and multiple Transformer encoder layers. We utilize overlapping patch embedding [7] to tokenize the audio spectrogram. Specifically, given an input spectrogram of size $H \times W \times 1$, we feed it to a convolution with the stride of S , the kernel size of $2S - 1$, the padding size of $S - 1$, and the kernel number of C_1 , so the output size is $\frac{H}{S} \times \frac{W}{S} \times C_1$. The Transformer encoder in the stage i has L_i encoder layers, each of which is composed of an attention layer and a convolutional feed-forward layer. Additionally, we introduce a multi-scale position encoding (MsPE) module, which incorporates the positional encoding within the self-attention operation and operates on projected values in each Transformer block [8]. MsPE naturally supports varying input sizes, and is thus especially effective and friendly for sound event detection and audio classification tasks.

2.2. Quadtree Attention

In our work, we use the Pyramid Vision Transformer (PVT) [9] backbone which applies spatial-reduction attention (SRA) to process features. SRA reduces the spatial scale of \mathbf{K} and \mathbf{V} before the attention operation, as demonstrated in the right of Figure 2. Although SRA can alleviate the cost increase in self-attention layers, its computation and memory complexity is still quadratic to the input size. Besides, SRA uses downsampled keys and values, which is harmful to capture frame-level details to some degree.

Inspired by the observation that not all patches are relevant in an audio spectrogram, we introduce quadtree attention (QA) to build token pyramids and compute attention in a coarse to fine manner [8]. For example, as illustrated in the middle of Figure 2, at the 1st stage, we compute the attention of the blue patches in spectrogram A with all the patches in spectrogram B and choose the top K (here, $K = 2$) patches highlighted in blue. In the 2nd stage, for the four framed sub-patches in spectrogram A (i.e., the children patches of the blue patch at the 1st

stage), we only compute their attentions with the sub-patches corresponding to the top K patches in spectrogram B at the 1st stage and other sub-patches are skipped to reduce computation. We highlight two sub-patches in spectrogram A in red and green with their corresponding top K patches in spectrogram B also highlighted in the same color. In this way, we can quickly skip irrelevant regions in the fine level if their corresponding coarse level regions are not promising. Therefore, our method can both obtain fine scale attention and retain long-range connections with low memory and computational costs. Specifically, given an input of size $h \times w \times c$, the computational complexities of SRA and QA are:

$$SRA : \mathcal{O}\left(\frac{h^2 w^2 c}{R^2} + h w c^2 R^2\right) \quad (1)$$

$$QA : \mathcal{O}(K h w c) \quad (2)$$

where R is the spatial reduction ratio of SRA and K is the number of selected tokens with the highest attention scores.

3. Proposed Method

3.1. Modeling WSSSED as Domain Adaption

Given an input audio spectrogram \mathbf{X} , a feature extractor $f(\cdot)$ and a classifier $c(\cdot)$ are learned to extract features and produce the prediction probability, respectively. As for WSSSED, only the clip-level classification label is available for the whole training process, so the pooling function $p(\cdot)$ is added to aggregate the frame-level feature into clip-level. To model WSSSED as the DA method, two feature sets $\mathbb{S} : \{s\}$ and $\mathbb{T} : \{t\}$ are constructed, where \mathbb{S} and \mathbb{T} represent the source (clip) domain and target (frame) domain for clarity. The corresponding label sets of \mathbb{S} and \mathbb{T} are defined as \mathbb{Y}^s and \mathbb{Y}^t , respectively. Under this view, WSSSED can be seen as training the classifier $c(\cdot)$ on the clip domain \mathbb{S} with label set \mathbb{Y}^s in a fully-supervised way, and then taking inference on the frame domain \mathbb{T} to estimate the frame-level label \mathbb{Y}^t in the testing process, as shown in Figure 1 B. This process is a typical setting of the DA task, which aims at learning a discriminative model in the presence of the feature shift between training and testing process [10]. Based on this perspective, the object of our DA-WSSSED pipeline can be formulated as:

$$L(\mathbb{S}, \mathbb{Y}^s, \mathbb{T}) = L_c(c(\mathbb{S}), \mathbb{Y}^s) + L_{dad}(\mathbb{S}, \mathbb{T}) \quad (3)$$

where $L_c(\cdot)$ is the classification loss that supervises the accuracy on the source domain.

$$L_c(c(\mathbb{S}), \mathbb{Y}^s) = \sum_{(s_i, y_i) \in (\mathbb{S}, \mathbb{Y}^s)} L_{ce}(c(s_i), y_i) \quad (4)$$

$L_{ce}(\cdot)$ denotes the cross-entropy loss. $L_{dad}(\cdot)$ is the domain adaption detection (DAD) loss to minimize the mismatch between \mathbb{S} and \mathbb{T} . The $L_{dad}(\cdot)$ loss forces $f(\cdot)$ and $p(\cdot)$ to learn domain-invariant features between the source (clip) domain and target (frame) domain, which helps the source trained classifier $c(\cdot)$ also perform well on target samples. Thus, the sound event duration estimates can be obtained more precisely in the testing process.

3.2. Domain Adaption Detection Loss

Drawing on the idea of weakly supervised object localization with domain adaption [6], there are also some properties for WSSD that do not exist in traditional DA tasks [11, 12], which poses a major obstacle for modeling WSSD as the DA task. Property 1 is that the target domain \mathbb{T} contains samples that do not belong to any classes of source domain \mathbb{S} , i.e., the background sound, so aligning features of these samples with the source domain will hurt the performance. For Property 2, the difference between the source domain and target domain is attributed to the pooling function $p(\cdot)$ rather than entirely unknown as traditional DA tasks. This property can be used as a prior when aligning the feature distribution of source and target domain. Property 3 stems from the fact that the number of samples in the source domain is much less than the target domain in WSSD. The insufficient samples cause difficulties when perceiving the source distribution during training.

Considering these properties, a domain adaption detection (DAD) loss is introduced as $L_{dad}(\cdot)$. We divide the target set \mathbb{T} into three subsets: 1) the Universum set $\mathbb{T}^u: \{t^u\}$ contains target samples whose label is unseen in source domain based on Property 1; 2) the fake target set $\mathbb{T}^f: \{t^f\}$ contains target samples that are highly correlated to the source domain as analyzed in Property 2; 3) the real target set $\mathbb{T}^t: \{t^t\}$ contains target samples that do not belong to \mathbb{T}^f and \mathbb{T}^u as discussed in Property 3. We use the target sampling strategy (TSA) [6] to select the representative samples of the three subsets from target features, which adopts the three-way K-means clustering method to assign frame-level features to three subsets. Based on the three subsets of the target domain, the DAD loss is defined to minimize the domain mismatch:

$$L_{dad}(\mathbb{S}, \mathbb{T}) = \lambda_1 L_{da}(\mathbb{S} \cup \mathbb{T}^f, \mathbb{T}^t) + \lambda_2 L_u(\mathbb{T}^u) \quad (5)$$

where λ_1, λ_2 are two parameters, $L_{da}(\cdot)$ is the domain adaption loss, and $L_u(\cdot)$ is the Universum regularization [13]. In detail, the domain adaption loss $L_{da}(\cdot)$ can be implemented as unsupervised domain adaption (UDA) [14] approaches to align the feature distributions between the two domains without accessing frame-level labels. Here, we adopt the maximum mean discrepancy (MMD) [14] as the UDA method:

$$L_{da}(\mathbb{S} \cup \mathbb{T}^f, \mathbb{T}^t) = -\frac{\sum 2 * h(s_i, t_j)}{|\mathbb{S} \cup \mathbb{T}^f| * |\mathbb{T}^t|} + \frac{\sum h(s_i, s_j)}{|\mathbb{S} \cup \mathbb{T}^f|^2} + \frac{\sum h(t_i, t_j)}{|\mathbb{T}^t|^2} \quad (6)$$

where $h(\cdot)$ is the gaussian kernel, $s_i, s_j \in \mathbb{S} \cup \mathbb{T}^f, t_i, t_j \in \mathbb{T}^t$. Adding $L_{da}(\cdot)$ can tighten the source domain and the target domain, making the classifier trained on source domain also perform better for the target samples. In addition, $L_u(\cdot)$ adopts the mechanism of Universum [13] that uses target samples with the source-unseen label \mathbb{T}^u to enhance the performance on the

Table 1: The mAP results on AudioSet evaluation set.

Method	#Params.	mAP
Baseline [15]	2.6M	0.314
DeepRes [24]	26M	0.392
PANN [18]	81M	0.434
PSLA ^P [19]	13.6M	0.444
AST [2]	87M	0.366
AST ^P [2]	87M	0.459
HTS-AT [4]	31M	0.453
HTS-AT ^P [4]	31M	0.471
APT-SRA	26M	0.458
APT-QA	25M	0.460
APT-SRA ^P	26M	0.475
APT-QA ^P	25M	0.476

target set. It is implemented as feature-based l_1 regularization:

$$L_u(\mathbb{T}^u) = \sum_{t_i^u \in \mathbb{T}^u} |t_i^u| \quad (7)$$

which minimizes the feature strength of Universum samples. What's more, this regularization also reduces the domain mismatch attributed to $p(\cdot)$, because it eliminates noises caused by Universum samples when generating source feature.

4. Experiments

In this section, we evaluate the performance of our method on three datasets: the audio classification on AudioSet [15]; the WSSD on DCASE2017 [16] and Urban-SED [17] datasets.

4.1. AudioSet Experiments

4.1.1. Dataset and Training Details

The AudioSet contains over two million 10-sec audio samples labeled with 527 sound event classes. In this paper, we follow the same training pipeline in [18] by using the full-train set (2M samples) to train our model and evaluate it on the evaluation set (22K samples). All samples are converted to mono as 1 channel by 32kHz sampling rate. We use 1024 window size, 320 hop size, and 64 mel-bins to compute STFTs and mel-spectrograms. As a result, the shape of the mel-spectrogram is (1000, 64). We set 4 network groups with 3, 4, 6, 3 transformer blocks respectively and the channel number in each stage is $C=64, 128, 320, 512$. For the quadtree attention, we set the number of selected regions with the highest attention scores $K=32$. We use the balance sampler [18, 19] and data augmentation strategies including time shifting [20], mix-up [21] and SpecAug [22]. The model is trained via the AdamW [23] optimizer with a batch size of 32. The starting learning rate is set to 1e-4 and the learning rate is cut into half every 30000 iterations after 60000 iterations. Following the standard evaluation pipeline, we use the mean average precision (mAP) to verify the classification performance on Audioset's evaluation set.

4.1.2. AudioSet Results

In Table 1, we compare our APT with different benchmark models including the latest HTS-AT, PANN, PSLA, AST and two self-ablated variations: (1) APT-SRA: audio pyramid transformer with spatial reduction attention; (2) APT-QA: audio pyramid transformer with quadtree attention. We denote our

Table 2: Performance comparison of APT variants and previous methods on DCASE2017.

Method	AT-F1	Seg-F1	Event-F1
DCASE2017 Baseline [16]	0.182	0.284	-
Ensemble-CNN [26]	0.526	0.555	-
CDur [20]	0.553	0.508	0.152
CNN-Transformer [27]	0.629	0.556	-
CNN-biGRU [27]	0.632	0.564	-
APT-SRA ^P	0.660	0.612	0.196
APT-QA ^P	0.668	0.612	0.205
APT-SRA-DA ^P	0.664	0.626	0.213
APT-QA-DA ^P	0.670	0.628	0.218

APT with ImageNet-pretraining as APT^P, and so on for other models. For the AST and HTS-AT, we compare our APT with their single models instead of the ensemble one to ensure the fairness of the experiment. We can see that our APT with SRA mechanism achieves a new SOTA mAP 0.475. When replacing SRA with QA, the performance is further improved to 0.476 with fewer parameters, which demonstrates the effectiveness of QA mechanism to a great extent. We also provide the mAP result of APT variants without pretraining for comparison. APT-QA without pretraining can achieve the mAP as 0.460, just 1.6% lower than 0.476. This indicates that our APT is not limited to the pretraining parameters of the computer vision model, and can be used flexibly in audio tasks.

4.2. DCASE2017 Task4 Experiments

4.2.1. Dataset and Training Details

The DCASE2017 task 4 – Large-scale weakly supervised sound event detection for smart cars dataset consists of a training subset with 51172 audio clips, a validation subset with 488 audio clips, and an evaluation set with 1103 audio clips, including 17 sound events. As the amount of data is much less than the full AudioSet, we just use 3 network groups with 3, 4, 6 pyramid-transformer blocks respectively. The latent dimension size is 64 and the final output latent dimension is 320. The two hyper-parameters of DAD loss are set as $\lambda_1 = 0.2$ and $\lambda_2 = 2$. The initial learning rate is set to $1e-4$ and the learning rate is cut into half if the validation loss does not improve for three epochs. We use Audio Tagging F1 score (AT-F1), Segment-F1 (Seg-F1) score and Event-F1 score [25] to evaluate our method. Specifically, AT-F1 measures the model’s capability to correctly identify the presence of an event within an audio clip. Seg-F1 can be seen as a coarse localization metric since precise timestamps are not required [20], while Event-F1 specifically describes a model’s capability to estimate the sound event duration.

4.2.2. DCASE2017 Task4 Results

For the WSSSED task, in addition to the above two APT variants: APT-SRA and APT-QA, we add two self-ablated variations which uses the proposed DA-WSSSED pipeline to replace the MIL-based WSSSED method, denoted as APT-SRA-DA and APT-QA-DA respectively, while other methods use the MIL-based WSSSED approach by default. As shown in Table 2, we can see that our APT-SRA significantly improves the Tag-F1 (0.660), Seg-F1 (0.612) and Event-F1 (0.196) performance, achieving new SOTAs on all metrics compared with previous methods. The application of QA mechanism further

Table 3: Performance comparison of APT variants and previous methods on weakly labeled Urban-SED.

Method	AT-F1	Seg-F1	Event-F1
Base-CNN [17]	-	0.560	-
Multi-Branch [28]	-	0.616	-
CDur [20]	0.771	0.647	0.217
APT-SRA ^P	0.786	0.666	0.224
APT-QA ^P	0.790	0.667	0.226
APT-SRA-DA ^P	0.793	0.678	0.236
APT-QA-DA ^P	0.796	0.682	0.239

enhances performance in terms of AT-F1 and Event-F1. Moreover, by replacing the MIL-based WSSSED approach with DA-WSSSED pipeline, our APT-QA-DA achieves significant performance boosts, especially on the Seg-F1 (0.628) and Event-F1 (0.218) metrics.

4.3. Urban-SED Experiments

4.3.1. Dataset and Training Details

Urban-SED is a sound event detection dataset within an urban setting, having 10 event labels. The Urban-SED dataset encompasses 10,000 soundscapes generated using the Scaper soundscape synthesis library, being split into 6000 training, 2000 validation and 2000 evaluation clips. For this dataset, we use two stages with 3, 4 pyramid-transformer blocks. The channel numbers in these two stages are 64 and 128, respectively.

4.3.2. Urban-SED Results

In Table 3, we compare previous approaches on the weakly labeled Urban-SED corpus to our APT variants. Our APT can be seen to outperform all compared approaches. The preeminent performances of APT on Urban-SED and DCASE2017 datasets demonstrate the effectiveness of our method on small datasets. Furthermore, benefited from eliminating the domain mismatch between training and testing process, our APT-QA-DA (0.796 AT-F1, 0.682 Seg-F1, 0.239 Event-F1) performs best among all models, which strongly proves the superiority of our DA-WSSSED pipeline.

5. Conclusions

In this paper, we first introduce an audio pyramid transformer with quadtree attention mechanism, which achieves new SOTAs on multiple sound event detection and audio classification datasets while reducing the computational complexity to linear. Furthermore, we provide a novel perspective that models WSSSED as a domain adaption task and introduce a DAD loss by analyzing the properties for WSSSED that do not exist in traditional DA tasks to align the feature distribution of source (clip) and target (frame) domain. Experiments show that our DA-WSSSED pipeline yields significant performance gains compared to the MIL-based WSSSED method.

6. Acknowledgements

This paper was partially supported by Shenzhen Science & Technology Research Program (No: GXWD20201231165807007-20200814115301001; No: JSGG20191129105421211) and NSFC (No: 62176008).

7. References

- [1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [2] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," *Interspeech*, 2021.
- [3] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 347–10 357.
- [4] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection," *arXiv preprint arXiv:2202.00874*, 2022.
- [5] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.
- [6] L. Zhu, Q. She, Q. Chen, Y. You, B. Wang, and Y. Lu, "Weakly supervised object localization as domain adaption," *arXiv preprint arXiv:2203.01714*, 2022.
- [7] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pvtv2: Improved baselines with pyramid vision transformer," *arXiv preprint arXiv:2106.13797*, 2021.
- [8] S. Tang, J. Zhang, S. Zhu, and P. Tan, "Quadtree attention for vision transformers," *arXiv preprint arXiv:2201.02767*, 2022.
- [9] Wang, Wenhai and Xie, Enze and Li, Xiang and Fan, Deng-Ping and Song, Kaitao and Liang, Ding and Lu, Tong and Luo, Ping and Shao, Ling, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 568–578.
- [10] C. Yu, J. Wang, Y. Chen, and M. Huang, "Transfer learning with dynamic adversarial adaptation network," in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019, pp. 778–786.
- [11] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.
- [12] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4893–4902.
- [13] J. Weston, R. Collobert, F. Sinz, L. Bottou, and V. Vapnik, "Inference with the universum," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 1009–1016.
- [14] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.
- [15] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.
- [16] 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 *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.
- [17] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 344–348.
- [18] 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.
- [19] Y. Gong, Y.-A. Chung, and J. Glass, "Psla: Improving audio tagging with pretraining, sampling, labeling, and aggregation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3292–3306, 2021.
- [20] H. Dinkel, M. Wu, and K. Yu, "Towards duration robust weakly supervised sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 887–900, 2021.
- [21] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.
- [22] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," *Interspeech*, 2019.
- [23] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.
- [24] L. Ford, H. Tang, F. Grondin, and J. R. Glass, "A deep residual network for large-scale acoustic scene analysis," in *Interspeech*, 2019, pp. 2568–2572.
- [25] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.
- [26] D. Lee, S. Lee, Y. Han, and K. Lee, "Ensemble of convolutional neural networks for weakly-supervised sound event detection using multiple scale input," *Detection and classification of acoustic scenes and events (DCASE)*, 2017.
- [27] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2450–2460, 2020.
- [28] Y. Huang, X. Wang, L. Lin, H. Liu, and Y. Qian, "Multi-branch learning for weakly-labeled sound event detection," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 641–645.