Feature-informed Embedding Space Regularization For Audio Classification

Yun-Ning Hung[1]
Speech, Audio & Music Intelligence Team
ByteDance
yunning.hung@bytedance.com

Alexander Lerch
Music Informatics Group
Georgia Institute of Technology
alexander.lerch@gatech.edu

Abstract—**Feature representations derived from models pre-trained on large-scale datasets have shown their generalizability on a variety of audio analysis tasks. Despite this generalizability, however, task-specific features can outperform if sufficient training data is available, as specific task-relevant properties can be learned. Furthermore, the complex pre-trained models bring considerable computational burdens during inference. We propose to leverage both detailed task-specific features from spectrogram input and generic pre-trained features by introducing two regularization methods that integrate the information of both feature classes. The workload is kept low during inference as the pre-trained features are only necessary for training. In experiments with the pre-trained features VGGish, OpenL3, and a combination of both, we show that the proposed methods not only outperform baseline methods, but also can improve state-of-the-art models on several audio classification tasks. The results also suggest that using the mixture of features performs better than using individual features.**

Index Terms—**Music auto-tagging, transfer learning, audio classification, regularization**

I. INTRODUCTION

Feature representations learned with deep architectures from large amounts of training data have shown to be powerful for a variety of tasks beyond the tasks they were trained for. The pre-trained features from the BERT model [1], for example, have been successfully used in multiple Natural Language Processing (NLP) tasks such as question answering, text classification, and language inference [2]. In the image domain, large-scale pre-trained models such as ImageNet [3] and VGG [4] have achieved competitive results in image classification and assist in many visual feature extraction tasks [5].

There exist several pre-trained models in the audio domain, such as L^3-Net [6], VGGish [7], and Jukebox [8]. L^3-Net leverages both audio and visual information provided by video to train an audio feature extraction network, while the VGGish model is pre-trained on a large-scale audio dataset, AudioSet [7]. Jukebox adopts a language model similar to NLP and is pre-trained on music audio only. Those features have been successfully applied to various Music Information Retrieval (MIR) tasks, including weakly-supervised instrument recognition [9], cross-modal representation learning [10], [11], music auto-tagging [12], [13], music emotion recognition [14], [15], and music genre classification [16], [17]. This successful

---

[1] This work was performed while Y. Hung was a master's student at Georgia Institute of Technology.

application to a multitude of different tasks combined with the comparably compact representation implies that these features are able to capture many task-agnostic properties of audio and music signals.

Although these transfer learning approaches are used successfully for such a wide range of tasks, there are potential drawbacks when using pre-trained features directly as the input representation. First, the extracted compact representation might lack task-specific information, which, in contrast, a task-specific representation learned from spectrogram input will include [18]. While it is possible to update and fine-tune the pre-trained model with task-specific datasets to catch this relevant information [19], the computational burden during training is increased due to the high complexity of the feature extractor. Second, the time resolution of pre-trained features is fixed, forcing any system utilizing the features to the same time resolution. VGGish features, for example, have a time resolution of approx. 1 s. Tasks that need a higher resolution such as beat-tracking [20], might not be able to utilize pre-trained feature representations. Third, extractors such as VGGish and L^3-Net rely on fixed input representations such as Mel spectrograms to extract features. Several recent studies, however, have shown that other input representations such as harmonic representations [21] or multi-rate PCEN [22] achieve superior performance on several audio-related tasks. Fourth, the deep architectures of pre-trained feature extractors also increase the computational workload and the execution time at inference. Small real-time devices with limited computational power, for example, might not be able to benefit from the pre-trained representations.

To address these drawbacks, we propose to incorporate the task-agnostic information of pre-trained representations into a new smaller-scale model without using them as input to the model. Our approaches show some similarity to feature-based knowledge distillation approaches used in teacher-student learning [23], where the pre-trained representations are used to regularize the embedding space during training. Thus, the pre-trained representations provide condensed, task-agnostic information to help shape the embedding, but are not required during inference. The architecture of the original model does not need to change, so that the input representation and its time resolution can be chosen freely for the task at hand. Such feature-based knowledge transfer approaches are rarely
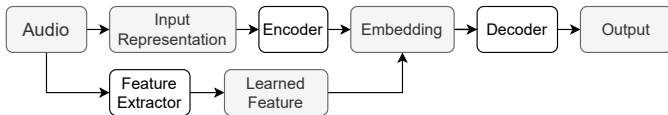
Fig. 1: Pipeline for the training including the learned feature. The lower branch is discarded during inference.

explored in the music/audio domain. Moreover, to extend commonly-used regularization loss functions (e.g., $\mathcal{L}_1$, $\mathcal{L}_{\text{CE}}$), we propose two loss functions based on cosine distance to match the embeddings. Cosine distance has been commonly used in several metric learning scenarios, such as few-shot learning [24], [25] and recommendation system [26].

To summarize, the main contributions of this paper are: (i) the introduction of two regularization methods integrating the information of pre-trained feature representations during training time without increasing the computational complexity during inference, and (ii) an investigation of the suitability of the two representations VGGish and OpenL3 and their combination for this regularization, and a detailed study of the proposed regularization methods for SOTA models.

II. METHODS

We propose two approaches to incorporate pre-trained feature representations into existing model architectures. In the following sections, we first introduce the general training pipeline of our regularization methods and the two pre-trained representations we use in our experiment. Then, details of two proposed regularization methods will be presented.

A. System Description

Figure 1 shows the general structure of our training pipeline. For the regularization to be applicable, we assume the deep-learning model to be composed of a similar pipeline with an encoder (E) to extract the intermediate representation (l) and a decoder (D) to predict the output based on this representation. The intermediate representation $l \in R^{T_{\text{E}} \times B_{\text{E}}}$ is also referred to as embedding, where T_{E} represents the number of time frames while B_{E} represents the dimensionality of the embedding.

The pre-trained features are a 2-dimensional representation $v \in R^{T_{\text{F}} \times B_{\text{F}}}$ where T_{F} represents the number of time frames while B_{F} represents the dimensionality of the learned features. Although we only investigate two widely used representations extracted with deep architectures, it is conceivable that any kind of custom feature in this format can be used.

B. Learned Features

We investigate three regularization inputs in this study, VGGish [7], OpenL3 [6], and the combination of both. VGGish and OpenL3 features are chosen due to their proven success in several audio-related downstream tasks [9], [11], [12], [16]. Both representations are extracted by models trained on large-scale audio datasets. VGGish features are extracted by the VGGish model [7] pre-trained on AudioSet [27] to perform audio classification at a time resolution of approx. 1 s (no

overlap) with a feature dimensionality of $B_{\text{F}} = 128$. The features are PCA transformed (with whitening) and quantized to 8-bits. The OpenL3 features [6] are extracted from L^3-net pre-trained on a subset of AudioSet. These features are also 2-dimensional representations with a time resolution of 0.1 s (no overlap) and $B_{\text{F}} = 512$.

The combination of these two features is achieved by a simple concatenation along the B_{F} dimension. To match the temporal resolution of OpenL3, VGGish features are repeated 10 times along the temporal dimension. The resulting combined representation has a time resolution of 0.1 s and a dimensionality of $B_{\text{F}} = 640$.[1]

C. Proposed Feature Integration Methods

Since VGGish, OpenL3, and the mixture features have different time resolution than the embeddings from most of the existing models, we repeat or average-pool the features n times with n representing the number of time frames in 1 s or 0.1 s to fit the time resolution. This approach is based on the assumption that a slight misalignment in time will not influence the result too much.

The design of the regularization methods is based on two assumptions. First, pre-trained features might contain information that is useful for various tasks but cannot be adequately represented in the unregularized embedding space (e.g., due to insufficient training data). Second, pre-trained features have strong discriminative power. The proposed regularization methods attempt to transfer the knowledge from the pre-trained feature space into the embedding vectors by adding structure to the embedding vectors resulting in a more separable embedding space.

Both methods add an extra loss term for network training:

$$L_{\text{final}} = L + \alpha L_{\text{Reg}}, \tag{1}$$

with the hyperparameter α adjusting the contribution on the overall loss L_{final}. The following methods are proposed to incorporate learned features into the embeddings:

- **Con-Reg**: Con-Reg aims at regularizing the embedding space so that its layout becomes more similar to the feature space of the pre-trained feature v. To do so, we utilize the features extracted from the audio and add an extra loss term $L_{\text{Con-Reg}}$ based on cosine distance[2] to minimize the distance between embeddings and pre-trained features:

$$L_{\text{Con-Reg}} = d_{\cos}(l, f(v)) \tag{2}$$

where f represents a 1D CNN with kernel size 1 [28] to transform the feature dimensionality B_{F} to match the embedding dimensionality, allowing us to compute the cosine distance between l and v.

- **Dis-Reg**: Dis-Reg is a distance-based regularization. Similar to *Con-Reg*, the embedding space is regularized with an additional loss. In this case, however, the additional

---

[1]More sophisticated interpolation methods for upsampling to match the time resolutions will be explored in future work.

[2]Pilot experiments using $\mathcal{L}_2$ as the distance function did not lead to competitive results.

Methods		DCASE 17 (F1)				MTAT (PR-AUC)			
		None	VGGish	OpenL3	Combined	None	VGGish	OpenL3	Combined
Baseline	Won et al. [21]	0.547	-	-	-	0.465	-	-	-
	Feature-only	-	0.496	0.477	0.501	-	0.454	0.454	0.456
	Concat	-	0.529	0.492	0.495	-	0.457	0.464	0.458
	FiLM	-	<u>0.548</u>	0.511	0.542	-	<u>0.470</u>	0.463	**0.470**
Proposed	Con-Reg	-	**0.568**	**0.557**	**0.576**	-	**0.471**	0.466	0.469
	Dis-Reg	-	<u>0.548</u>	0.543	<u>0.563</u>	-	0.464	**0.468**	0.463

TABLE I: Evaluation results for the DCASE 17 and MTAT datasets. Bold numbers represent the highest score in each category while the underlined numbers indicate the scores improved over the baseline model.

loss term $L_{\text{Dis}-\text{Reg}}$ aims at forcing the distances between pairs of embedding vectors to be similar to the distance of corresponding pairs of pre-trained features:

$$L_{\text{Dis}-\text{Reg}} = |d_{\cos}(l_i, l_j) - d_{\cos}(v_i, v_j)|, \qquad (3)$$

where $d_{\cos}$ represents the cosine distance between two embeddings from samples i and j ($i \neq j$) or the distance of two corresponding learned features, respectively.

D. Model Architecture

The harmonic CNN model proposed by Won et al. [21] is chosen as our experimental system. This model utilizes learnable harmonic filters to capture the inherent harmonic structure of the input audio and achieves SOTA results on music tagging, sound event tagging, and keyword spotting. The model is composed of seven residual blocks as an encoder to extract embeddings from the harmonic representation. Two linear layers are then used as a decoder to predict the output from the embeddings.

III. EXPERIMENT

A. Baseline Feature Integration Methods

To provide a baseline reference as a comparison to the proposed regularization methods, we also present the result for three baseline methods incorporating pre-trained features:

- **Features-only**: The simplest way of incorporating pre-trained features is to directly use these pre-trained features as the input for the decoder. The decoder of the original model is adjusted to fit the dimensionality of the features.
- **Concat**: Concat simply concatenates embeddings l and pre-trained features v along the B dimension. The pre-trained features, therefore, act as a supplementary of the embeddings and the embedding space is transformed into a joint feature space. The decoder then has the flexibility to decide how to leverage the information from the joint feature space. Since the concatenated embeddings have a larger dimension along B than the original embeddings, the first layer of the decoder will be adjusted to fit the dimensionality of the new intermediate representation.
- **FiLM**: A Feature-wise Linear Modulation (FiLM) layer is originally proposed as a general-purpose conditioning method to assist visual reasoning, which is difficult for standard deep-learning methods [29]. The success of

FiLM leads to its successful usage in several audio-related tasks [30], [31]. The FiLM layer influences neural network computation via a simple, feature-wise affine transformation based on conditioning information:

$$l_{new} = \gamma \cdot l + \beta \qquad (4)$$

where $\gamma = f(v)$ and $\beta = h(v)$ are calculated based on pre-trained features v. We use a simple linear layer for the functions f and h to compute the transformation parameters γ and β. The resulting embeddings after the transformation are l_{new}.

B. Tasks

We investigate our proposed methods on two different tasks to demonstrate the influence of the regularization: music tagging and sound event tagging as also used by Won et al. [21].

1) Music Auto-Tagging: Music auto-tagging is a multi-label classification problem aiming to predict tags for a musical piece. The used dataset is MagnaTagATune (MTAT) [32]. This dataset has approx. 21k audio clips with each clip around 30 s. The dataset contains a variety of tags, including genre, mood, and instrumentation. The top-50 tags are chosen for label prediction. The results are reported using the Area Under Precision-Recall Curve (PR-AUC). The OpenL3 features are extracted in the music subset mode.

2) Sound Event Tagging: Sound event tagging is an important task that has been included in DCASE challenges for many years [33]. The goal of this task is to detect audio events in a sound excerpt. Following Won et al., we choose "Task 4: Large-scale weakly supervised sound event detection for smart cars" from the DCASE2017 challenge as our target task. The dataset is a subset of AudioSet [27] and contains approx. 53k audio excerpts for 17 classes. Each excerpt is around 10 s. The results are reported using the average of instance-level F1-scores with a threshold value of 0.1. In this case, the OpenL3 features are extracted in the environmental subset mode.

C. Experimental Setup

The training setup, including dataset cleaning and split and the selection of optimizer and learning rate, is mirrored from their work. All methods are evaluated on each task by using VGGish, OpenL3, and the combined pre-trained features. The best model is selected based on the validation metric for each task for testing. After hyperparameter tuning, we choose $\alpha = 5$ for $L_{\text{Con}-\text{Reg}}$ and $\alpha = 1$ for $L_{\text{Dis}-\text{Reg}}$.
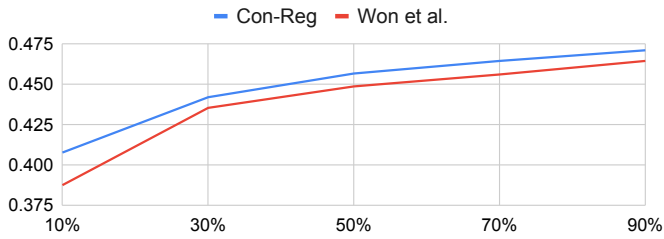
Fig. 2: Ablation study for the results under different percentage of training data for SOTA system and our best method.

Methods	Training Parameters			Inference Parameters		
	VGG	L3	Comb.	VGG	L3	Comb.
Feature-only	38k	136k	169k	72.2M	4.8M	77.0M
Concat	3.7M	3.8M	3.8M	75.8M	8.4M	80.6M
FiLM	3.7M	3.9M	4.0M	75.8M	8.6M	80.8M
Con-Reg	3.7M	3.8M	3.8M	3.6M	3.6M	3.6M
Dis-Reg	3.6M	3.6M	3.6M	3.6M	3.6M	3.6M

TABLE II: Number of training and testing parameters for both proposed and baseline methods.

IV. RESULT AND DISCUSSION

The experimental results are shown in Table I. In the following, we discuss different aspects of these results separately

1) Learned Features: It can be observed that VGGish features generally outperform the OpenL3 features. The combined features tend to perform on par or better than the individual features. This result suggests that —although VGGish and OpenL3 were both trained on AudioSet— they encode somewhat complementary information. This is most likely due to different training strategies and hints at the possibility of combining them for other downstream tasks.

2) Baselines: Among all the baseline methods, *FiLM* and *Concat* generally perform better than using only pre-trained features for prediction, but comparable to or only slightly better than Won et al.'s original system. This result indicates that there is some task-specific information that exists in the harmonic representations but not in the pre-trained features. *FiLM*, in general, tends to outperform *Concat*.

3) Proposed Methods: Both proposed methods tend to generally give higher results than the baselines and thus demonstrate the efficiency of our proposed regularization methods. Since the pre-trained features could not achieve better performance than learning from harmonic representation, directly incorporating features into embedding can potentially be seen as noise during inference and drop the performance. In contrast, indirectly leveraging pre-trained features does not have this limitation. The model is able to combine the task-specific information learned through the harmonic representation with the complementary, task-agnostic but highly meaningful information of the pre-trained features.

Con-Reg generally outperforms *Dis-Reg*. We discover that $L_{\mathrm{Dis-Reg}}$ has larger values than $L_{\mathrm{Con-Reg}}$ during training. The larger loss term indicates that optimizing the model to predict embedding vectors matching the distance of pre-trained features is harder than directly minimizing the distance between embedding vectors and pre-trained features. As a result, this regularization method could not structure embeddings space better than *Con-Reg*.

Furthermore, *Con-Reg* achieves better results than the SOTA model proposed by Won et al. [21] on both the DCASE 17 dataset and MTAT dataset.

4) Amount of Training Data: We can observe that the improvement for the MTAT dataset is very limited compared to

the DCASE 17 dataset. This might be due to the fact that the MTAT dataset is larger in size; therefore, the model is able to learn good embeddings directly from the training data and the additional information from the learned features is unnecessary. To compare the effect of insufficient training data, we present the results of an ablation study by randomly choosing x percent of audio files in the training set to form training subsets with $x \in 10\%, 30\%, 50\%, 70\%, 90\%$. We use the best setting, *Con-Reg* with combined features, for the experiment and compare the performance with the SOTA system. The results are shown in Figure 2. We can observe that with decreasing training data, the improvement from the proposed regularization method becomes more obvious, especially lower than 30% of training data.

A. Model Complexity

Table II shows the number of training and inference parameters for both the proposed and the baseline methods. Since audio files can be pre-processed to extract the learned features, the *feature-only* method only needs the decoder during training and has, therefore, the smallest number of trainable parameters. In contrast, the same *feature-only* method requires a large number of parameters during inference due to the large-scale VGGish/OpenL3 extractor. A similar imbalance between the number of training and inference parameters can be observed for the *Concat* and *FiLM* methods.

Comparing the baseline results from Table I with the number of baseline parameters listed in Table II we can conclude that the better the result, the more parameters the extractor needs. For example, VGGish features achieve better performance than OpenL3 but its extractor is also more complex. As mentioned above, the combination of pre-trained features can achieve better performance, but the parameters for inference increase considerably as they require two extractors (both VGGish and OpenL3) during inference.

The proposed regularization methods have a similar number of parameters as the original model proposed by Won et al. [21] (approx. 3.6M), for both training and inference. This is because our proposed methods only use features in the additional regularization loss function without modifying the model architecture.

V. CONCLUSION

In this work, we proposed two novel regularization methods to incorporate the information from pre-trained features during

training by adding an additional loss term restructuring the embedding space. The proposed regularization methods show improvements compared to baseline feature integration methods, which either directly use pre-trained features or directly influence the model through concatenation or a feature-wise linear transform. Furthermore, the regularized models can outperform SOTA audio classification models, especially having a more pronounced performance increase in the case of limited training data.

Future work will include experiments on other features, such as Jukebox [8] and Musicnn [34], as well as the exploration of different interpolation methods to deal with the various time resolutions. We also plan to extend our experiments to non-audio-related fields such as NLP and computer vision.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186.

[2] T. Falke, L. F. Ribeiro, P. A. Utama, I. Dagan, and I. Gurevych, "Ranking generated summaries by correctness: An interesting but challenging application for natural language inference," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 2214–2220.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[6] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 3852–3856.

[7] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. Saurous, B. Seybold, M. Slaney, R. A. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[8] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020.

[9] S. Gururani, M. Sharma, and A. Lerch, "An attention mechanism for musical instrument recognition," in *Proceedings of International Society for Music Information Retrieval Conference*, 2019, pp. 83–90.

[10] Y. Yu, S. Tang, F. Raposo, and L. Chen, "Deep cross-modal correlation learning for audio and lyrics in music retrieval," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 15, no. 1, pp. 1–16, 2019.

[11] D. Zeng, Y. Yu, and K. Oyama, "Audio-visual embedding for cross-modal music video retrieval through supervised deep CCA," in *International Symposium on Multimedia*, 2018, pp. 143–150.

[12] K. Koutini, H. Eghbal-zadeh, and G. Widmer, "Receptive field regularization techniques for audio classification and tagging with deep convolutional neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1987–2000, 2021.

[13] R. Castellon, C. Donahue, and P. Liang, "Codified audio language modeling learns useful representations for music information retrieval," in *Proceedings of International Society for Music Information Retrieval Conference*, 2021.

[14] E. Koh and S. Dubnov, "Comparison and analysis of deep audio embeddings for music emotion recognition," *AAAI Workshop on Affective Content Analysis*, 2021.

[15] S. Amiriparian, M. Gerczuk, E. Coutinho, A. Baird, S. Ottl, M. Milling, and B. Schuller, "Emotion and themes recognition in music utilising convolutional and recurrent neural networks," in *CEUR Workshop Proceedings*, vol. 2670, 2019.

[16] J. Ramírez and M. J. Flores, "Machine learning for music genre: multifaceted review and experimentation with audioset," *Journal of Intelligent Information Systems*, vol. 55, no. 3, pp. 469–499, 2020.

[17] Q. Huang, A. Jansen, L. Zhang, D. P. Ellis, R. A. Saurous, and J. Anderson, "Large-scale weakly-supervised content embeddings for music recommendation and tagging," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 8364–8368.

[18] D. Tompkins and E. Nichols, "DCASE 2019 challenge task 5: CNN+ vggish," *Technical Report, DCASE2019 Challenge*, 2019.

[19] Z. Xu, P. Smit, and M. Kurimo, "The Aalto system based on fine-tuned audioset features for dcase2018 task2 - general purpose audio tagging," in *Technical Report, DCASE2018 Challenge*, 2018.

[20] M. E. Davies, N. Degara, and M. D. Plumbley, "Evaluation methods for musical audio beat tracking algorithms," *Queen Mary University of London, Centre for Digital Music, Tech. Rep. C4DM-TR-09-06*, 2009.

[21] M. Won, S. Chun, O. Nieto, and X. Serra, "Data-driven harmonic filters for audio representation learning," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 536–540.

[22] C. Ick and B. McFee, "Sound event detection in urban audio with single and multi-rate PCEN," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 880–884.

[23] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[24] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[25] S. Gidaris and N. Komodakis, "Dynamic few-shot visual learning without forgetting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4367–4375.

[26] C. Hansen, C. Hansen, L. Maystre, R. Mehrotra, B. Brost, F. Tomasi, and M. Lalmas, "Contextual and sequential user embeddings for large-scale music recommendation," in *Proceedings of ACM Conference on Recommender Systems*, 2020, pp. 53–62.

[27] 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 *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 776–780.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[29] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[30] O. Slizovskaia, L. Kim, G. Haro, and E. Gomez, "End-to-end sound source separation conditioned on instrument labels," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 306–310.

[31] J. W. Kim, R. Bittner, A. Kumar, and J. P. Bello, "Neural music synthesis for flexible timbre control," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 176–180.

[32] E. Law, K. West, M. I. Mandel, M. Bay, and J. S. Downie, "Evaluation of algorithms using games: The case of music tagging." in *Proceedings of International Society for Music Information Retrieval Conference*, 2009, pp. 387–392.

[33] 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.

[34] J. Pons and X. Serra, "musicnn: Pre-trained convolutional neural networks for music audio tagging," in *Proceedings of International Society for Music Information Retrieval Conference*, 2019.