Automatic TV Genre Classification Based on Visually-Conditioned Deep Audio Features

Alessandro Ilic Mezza
Politecnico di Milano
Milan, Italy
alessandroilic.mezza@polimi.it

Paolo Sani
Politecnico di Milano
Milan, Italy
paolo1.sani@mail.polimi.it

Augusto Sarti
Politecnico di Milano
Milan, Italy
augusto.sarti@polimi.it

***Abstract*—Reliable tools for automatic genre classification (AGC) are highly sought-after by the television industry for the promise of saving the cost of manually annotating the ever-growing catalogs of media content providers. Metadata are indeed vital for a variety of tasks, including data analytics, database navigation, and recommender systems. In recent years, however, only a few works have focused on TV genre classification, possibly due to the lack of publicly available datasets of broadcast media. To bridge this gap and foster future research, we present ITTV, a manually annotated dataset of Italian TV programs gathered on YouTube. From this, we propose a novel AGC method based on deep audio features that rely on the well-established "Look, Listen and Learn" paradigm. Evaluated on ITTV, the proposed method is shown to provide state-of-the-art results, outperforming recent audio-based AGC methods.**

***Index Terms*—Automatic genre classification, TV programs, deep audio features, L^3-net.**

I. Introduction

With dozens of new programs broadcast daily, the online catalogs of television networks and multimedia streaming platforms are constantly growing and changing. To simplify content management, enable user navigation, collect analytics on users' behavior, and provide targeted recommendations based on individual preferences, every item in these catalogs must be assigned reliable metadata. However, manually annotating massive corpora of media content is expensive and could introduce labeling errors. In the past few years, these issues have motivated abundant research focused on developing automatic annotation tools for broadcast media [1]–[12].

Various works on automatic genre classification (AGC) report how combining complementary features from different modalities has been effective in improving the performance of multimedia classification systems [1], [2], [4]–[8]. In particular, [7] and [5] investigate the combination of text and audio features and of visual and audio features, respectively. Doulaty et al. [7] also report that adding auxiliary metadata, such as TV channel and time of broadcast, further improves the results.

The remarkable performance of multimodal AGC methods should come as no surprise, as such systems operate in a very similar way as humans do when it comes to discerning multimedia content. Nevertheless, collecting and combining data from different modalities is expensive in terms of computation and storage. Furthermore, findings obtained with a single modality have high research value, as they could eventually complement and improve multimodal approaches. For these reasons, this work focuses on audio-based AGC of TV programs, a less studied task than, for example, its video-based counterpart. After all, audio requires only a fraction of the memory footprint of video data and, unlike text, is readily available without further processing steps that could introduce transcription errors, especially for those languages that are less represented in natural language processing studies.

Earlier work on audio-based AGC was mainly based on statistical machine learning [9]–[11]. For instance, Ekenel et al. [9] extracted mel-frequency cepstral coefficients, fundamental frequency, signal energy, and zero-crossing rate from short-time audio segments. Then, they modeled each program as a Gaussian mixture model (GMM) whose parameters were fed to a support vector machine (SVM) classifier. Doulaty et al. [10] and Kim et al. [11] considered each TV program as a bag of "acoustic words" defined as the indices of the Gaussian components of a GMM or those of a dictionary obtained through the Linde-Buzo-Gray quantization algorithm, respectively. Hence, [10], [11] used latent Dirichlet allocation (LDA) followed by an SVM classifier. Furthermore, [7] used information regarding the most likely LDA latent topic to augment the vector of acoustic features and form the input to an artificial neural network classifier.

On the whole, the field of audio-based AGC seems to have been largely unaffected by the major deep learning advancements of the past decade. In fact, recent literature fundamentally lacks any study on classic deep learning architectures such as, e.g., end-to-end convolutional neural networks (CNN) that have instead become widespread in countless other audio classification tasks. Recently, however, Pham et al. [12] proposed to represent each TV program as a histogram of sound events. After windowing the input program, each segment is fed to a frozen audio-tagging pretrained audio neural network (PANN) [13] that outputs posterior probabilities for 527 sound categories from the AudioSet ontology [14]. Then, the κ most probable sound events are hot encoded, and the resulting vectors are accumulated along the temporal dimension and normalized with respect to the total number of audio tags (a technique that the authors call *mean-num-*κ) [12]. Finally, the feature vectors thus obtained are fed to a downstream classifier. This AGC method is reported to achieve state-of-the-art results on a private dataset of 6160 BBC programs.

Nevertheless, the shortcomings of a purely sound-event-based approach lie in the fact that if two genres happen to share a large number of characteristic sound sources, their histogram representations will end up being similar, leading to confusion. To overcome this potential problem, we propose to use audio features conditioned to implicitly incorporate visual information. Ultimately, this choice is motivated by the fact that human viewers can often distinguish acoustically similar TV shows by exploiting visual cues. In order to gather visually related information having access to audio data only, we adopt the "Look, Listen and Learn" (L^3) learning paradigm [15]–[17]. L^3-net is a neural architecture trained using an audio-visual correspondence (AVC) objective. Namely, two distinct subnetworks produce audio and visual embeddings, respectively, which are then passed to a stack of fusion layers that try to determine whether audio and video correspond or not. Once trained, each subnetwork can be used as a feature extractor independently of the other. In this work, we are only interested in audio-based classification, and the visual subnetworks are discarded after the AVC pretraining. Nevertheless, we leverage the fact that, thanks to AVC, the audio subnetworks were encouraged to capture latent semantic aspects carried by the visual modality. In particular, we propose the use of a parallel arrangement of specialized L^3-net audio embedding extractors driving an early program-wise aggregation stage prior to the downstream TV genre classifier.

The contribution of this work is threefold.

1) We present ITTV, a manually annotated dataset of over 675 hours of Italian television collected from YouTube.
2) We study the performance of end-to-end CNN classifiers with late majority-voting decision, a family of baseline systems that was previously missing from the literature on audio-based AGC.
3) We propose a novel two-stage audio-based AGC algorithm based on features extracted using frozen L^3 audio subnetworks.

The remainder of the manuscript is organized as follows. In Section II, we present the ITTV dataset. In Section III, we describe the proposed AGC method. In Section IV, we outline the experimental setup and discuss the results. In Section V, we conduct an ablation study to support our design choices. Finally, Section VI concludes this work.

II. ITTV Dataset

To the best of our knowledge, no dataset of TV programs is currently freely available for research purposes. Indeed, most datasets are proprietary to national television networks and only accessible upon request, such as those in [4], [12], while others were made temporarily available to the participants of a challenge, such as [18].

In this paper, we present ITTV, a manually annotated dataset of over 675 hours of YouTube clips of Italian TV programming. The dataset is structured as follows: the proposed training set contains 1575 clips, whereas the validation and test sets comprise 525 clips each. Similarly to the RAI dataset described in [4], each clip is annotated with one of seven categories, i.e., *Cartoons*, *Commercials*, *Football*, *Music*, *News*, *Talk Shows*, and *Weather Forecast*. Compared to the RAI dataset [4] that includes 262 programs totaling 110 hours, ITTV is about six times larger. Moreover, ITTV is balanced across classes, meaning that each genre is represented by 375 clips, 225 of which are in the proposed training set, 75 in the proposed validation set, and 75 in the proposed test set. As in [4], we favor full-length TV programs, ranging from hours-long talk shows to few-second commercials, and manually assign a single categorical label to each installment. Table I reports the duration of the clips for each genre. Notably, since their length varies widely, balancing with respect to the number of clips does not correspond to balancing with respect to the time duration of each class. Training and validation sets contain different installments of the same programs (e.g., different episodes of a given animated series), with the exception of *Commercials* for which a similar intra-installment relationship cannot be established. The test set, instead, is disjoint from the training set and contains programs from different television broadcasters. In order to account for different levels of audio and video quality, the dataset includes programs from national television networks and local TV stations. In doing so, official YouTube channels have been preferred to minimize the risk of video takedowns. ITTV is made publicly available online.[1]

TABLE I
AN OVERVIEW OF THE ITTV DATASET.

Genre	Train	Validation	Test	Total
Cartoons	87 h 46 min	31 h 5 min	24 h 51 min	143 h 42 min
Commercials	1 h 20 min	25 min	27 min	2 h 12 min
Football	12 h 0 min	3 h 53 min	7 h 4 min	22 h 57 min
Music	10 h 22 min	3 h 16 min	6 h 9 min	19 h 47 min
News	107 h 9 min	38 h 0 min	26 h 14 min	171 h 23 min
Talk Shows	98 h 59 min	31 h 55 min	171 h 58 min	302 h 52 min
Weather Forecast	7 h 15 min	1 h 35 min	4 h 15 min	13 h 5 min
Total	**324 h 51 min**	**110 h 9 min**	**240 h 58 min**	**675 h 58 min**

III. Proposed Method

The proposed AGC method, illustrated in Fig. 1, consists of two stages: (i) deep-learning-based feature extraction with early (nonlinear) aggregation and (ii) downstream classification. In this work, we use a parallel combination of two pretrained L^3 audio subnetworks made available by the authors of [17], each having 4.68 M trainable parameters. These embedding extractors are trained using two distinct subsets of AudioSet [14]: an environmental subset (*env*) and a music-oriented subset (*music*) [17]. By combining these two data representations, we expect to encompass the majority of the most common sounds that may occur in a television program.

First, an entire input program is segmented into K non-overlapping frames of L samples. Namely, the kth frame can be expressed as

$$\mathbf{x}_k = [x[kL], x[kL+1], ..., x[(k+1)L-1]]^{\mathrm{T}} \qquad (1)$$

[1] [Online] Available: https://zenodo.org/record/8027327

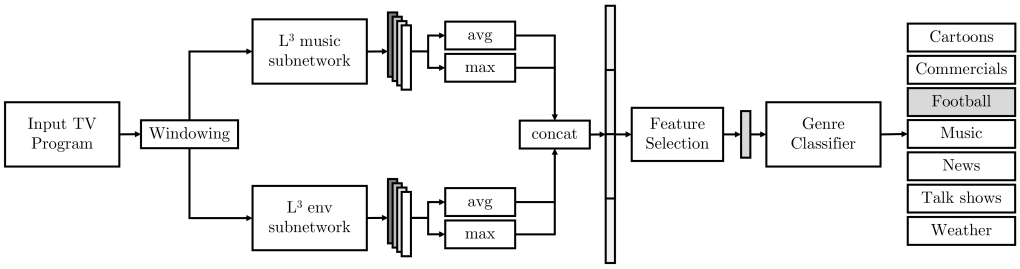


Fig. 1. Proposed method. In the scheme, the two L^3 audio subnetworks (*music* and *env*) are frozen.

where $k = 0, ..., K-1$. Each audio frame in a program is then fed to both the *env* and *music* L^3-nets yielding the embeddings $\mathbf{e}_k \in \mathbb{R}^N$ and $\mathbf{m}_k \in \mathbb{R}^N$, respectively. Hence, we aggregate both types of embeddings across all K frames. To do so, we compute average- and max-pooling along the temporal dimension to obtain two global embeddings for each type. More specifically, as far as music embeddings are concerned, we compute $\mathbf{m}_{\text{avg}} = \frac{1}{K}\sum_{k=0}^{K-1}\mathbf{m}_k$ and $\mathbf{m}_{\text{max}} = [\bar{m}_1, ..., \bar{m}_N]^{\text{T}}$ where

$$\bar{m}_j = \max\{\mathbf{m}_k[j] : k = 0, ..., K-1\}. \quad (2)$$

Correspondingly, the same is done to obtain $\mathbf{e}_{\text{avg}} \in \mathbb{R}^N$ and $\mathbf{e}_{\text{max}} = [\bar{e}_1, ..., \bar{e}_N]^{\text{T}}$. We then construct a compound representation by concatenating $\mathbf{m}_{\text{avg}}$, $\mathbf{m}_{\text{max}}$, $\mathbf{e}_{\text{avg}}$, and $\mathbf{e}_{\text{max}}$ in a single feature vector $\mathbf{f} \in \mathbb{R}^{4N}$ representing the entire program. Being derived from the raw activations of the fully-connected output layers of two deep neural networks, $\mathbf{f}$ is a high-dimensional dense data representation that is likely to contain redundant information. Therefore, we apply discriminative dimensionality reduction via a feature selection algorithm $\varphi : \mathbb{R}^{4N} \rightarrow \mathbb{R}^M$. Finally, we feed the M selected features to a downstream genre classifier that yields a posterior probability for each class in the RAI taxonomy [3].

IV. Evaluation

A. Baseline Systems

The method proposed by Pham et al. [12] can be regarded as state of the art for audio-based AGC. It uses an audio-tagging PANN model to predict the sound events occurring in every non-overlapping 5-second frame extracted from a TV program. First introduced in [13], PANNs are VGG-like architectures trained on AudioSet [14]; our implementation of [12] uses the pretrained CNN14 model released by the authors of [13].

Additionally, we evaluated several end-to-end CNN architectures with late majority voting. First, we resampled every TV program at 32 kHz, and windowed them into non-overlapping frames of 5 seconds. We then assigned the label of the corresponding TV program to each audio frame. For each frame, we extracted the log-amplitude mel-frequency spectrogram with minimum frequency at 50 Hz, maximum frequency at 14 kHz, and window size $L = 1024$. The number of mel bands B and the hopsize R were selected according to the input size of the target CNN model. At inference time, the decision on the entire TV program was taken by applying soft majority voting to the class posteriors across all frames.

All end-to-end baselines gave similar results, regardless of the specific architecture. Therefore, for the sake of brevity, we report the results of two exemplary networks, both modified to include a 7-way output softmax layer: EfficientNet-b0 [19] ($R = 512$; $B = 224$) and CNN14 [13] ($R = 320$; $B = 64$). The former was chosen for being a well-established lightweight model suited for mobile devices, whereas the latter offers a direct comparison with the AGC method in [12].

B. Downstream Classification

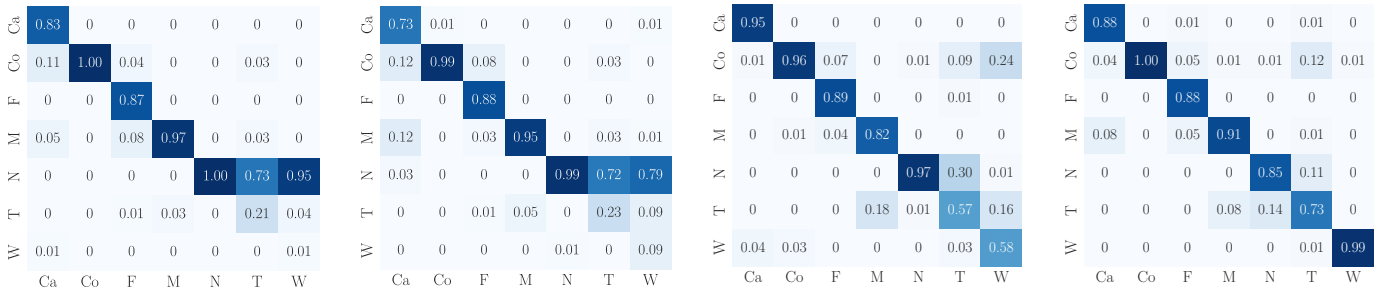
Both [12] and our AGC method decouple feature extraction and aggregation from the genre classifiers. As in [12], in order to highlight the contribution of the feature extraction stage, both methods use a simple random forest (RF) with 200 decision trees as genre classifier.

In our implementation, we set the L^3 embedding size and the number of selected features to $N = 6144$ and $M = 100$, respectively. This corresponds to a dimensionality reduction by a factor of about 245, operated by the feature selection algorithm φ. Despite being independent of the choice of the subsequent classifier, we opted for an algorithm that is synergistic with the RF. Namely, we select the M features with highest Gini importance (also known as mean decrease in impurity [20]) computed on the entire training dataset.

C. Results and Discussion

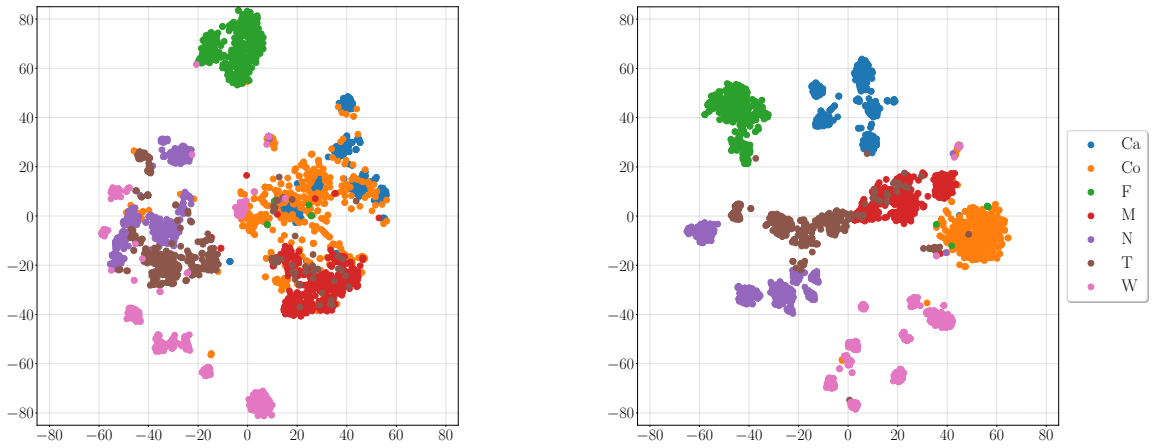
Fig. 2 shows the confusion matrices of the four methods considered in the present study. In Fig. 2a, EfficientNet-b0 is shown to achieve a classification accuracy of 69.9%. Similarly, in Fig. 2b, CNN14 achieves an accuracy of 69.3%. In Fig. 2c, Pham et al. [12] is shown to outperform both end-to-end architectures with an average accuracy of 82.1%. Finally, in Fig. 2d, our method provides the overall best classification results, with an accuracy of 89.01%.

On the one hand, all four models are able to classify four out of the seven genres with high accuracy, i.e., *Cartoons*, *Commercials*, *Football*, and *Music*. This is possibly due to the fact that these genres have distinct acoustic signatures and are generally easier to discriminate using audio cues. On the other hand, *News*, *Talk Shows*, and *Weather Forecast* appear to be much more difficult to recognize. In particular, both CNN14 and EfficientNet-b0 erroneously attribute most *Talk Shows* and *Weather Forecast* samples to the class *News*. This confusion, indeed, seems to be the main culprit for the poor



(a) EfficientNet-b0 (b) CNN14 (c) Pham et al. [12] (d) Proposed method

Fig. 2. Normalized confusion matrices for *Cartoons* (Ca), *Commercials* (Co), *Football* (F), *Music* (M), *News* (N), *Talk Shows* (T), and *Weather Forecast* (W).



(a) Pham et al. [12] (b) Proposed method

Fig. 3. t-SNE visualization of the feature vectors extracted by (a) Pham et al. [12] and (b) the proposed method; best viewed in color.

performance of end-to-end networks with late majority voting. Interestingly, this issue arises regardless of the model capacity: with 14 M parameters, EfficientNet-b0 is significantly smaller than CNN14, which has just over 80.75 M parameters; yet the two networks perform almost identically.

Conversely, Pham et al. [12] proves capable of discriminating *News*, *Talk Shows*, and *Weather Forecast* to a greater extent. In turn, this suggests that creating a full-program representation via early feature aggregation, i.e., prior to the downstream classifier, is beneficial for AGC as it allows to nonlinearly capture the importance of rare and yet informative audio frames. Nevertheless, *Talk Shows* and *Weather Forecast* still exhibit the lowest classification accuracy among all classes, both being below 60%. These genres, along with *News*, are all characterized by similar sounds, i.e., mostly human speech, a fact that may constitute a challenge for [12] whose final decision is based on histograms of sound events.

Taking advantage of the assumption that similar-sounding programs may still differ when considering visual cues, the proposed method provides a significant improvement over [12], especially concerning the three speech-centered classes for which it achieves an average absolute increment of +15%. Indeed, of the three classes, none is classified with less than 73% accuracy, as shown in Fig. 2d. Notably, these improvements are obtained using the proposed parallel subnetwork arrangement described in Section III, totaling 9.37 M trainable parameters, i.e., considerably less than the 80.75 M of the CNN14 model [13] used in [12].

In order to better understand the results, Fig. 3 reports the t-SNE of the feature vectors extracted by Pham et al. [12] and by our method. In particular, Fig. 3b shows that the L^3-net embeddings of classes *News* (in purple), *Talk Shows* (in brown), and *Weather Forecast* (in pink) form separable and identifiable clusters, contrarily to what happens with the corresponding sound-event-based features depicted in Fig. 3a.

V. Ablation Study

In Section III, we proposed a method consisting of a parallel configuration of *env* and *music* L^3-nets, each followed by a parallel average- and max-pooling aggregation. In this section, we conduct an ablation study comparing the results obtained using only the *env* or the *music* subnetwork, as well as applying only average- or max-pooling to aggregate the resulting L^3 embeddings. The results are shown in Table II, where the symbol $\|$ indicates the parallel arrangement of two modules. All results are obtained by fixing $M = 100$. For completeness, the methods based on L^3-net are compared with the baseline system by Pham et al. [12].

TABLE II
CLASSIFICATION ACCURACY (%) OF THE MODELS CONSIDERED IN THE ABLATION STUDY EVALUATED ON THE ITTV TEST SET.

Model	Aggregation				Parameters
	mean-num-10	avg	max	avg ‖ max	
Pham et al. [12]	82.10	—	—	—	80.75 M
L^3 env	—	72.83	75.14	86.51	4.68 M
L^3 music	—	67.24	73.99	87.28	4.68 M
L^3 music ‖ L^3 env	—	73.02	74.18	**89.01**	9.37 M

Of the two aggregation functions applicable to L^3 embeddings, max-pooling (fourth column) consistently outperforms average-pooling (third column), suggesting that it is important to nonlinearly capture the activation of rare audio frames when creating a full-program representation. However, when both aggregation methods are used in parallel (fifth column), the results improve remarkably, leading to an absolute increment ranging from $+11.4\%$ using the *env* subnetwork to $+14.8\%$ using the parallel configuration of both L^3-nets.

Using parallel aggregation, the *music* subnetwork (third row) performs marginally better than the *env* subnetwork (second row), a behavior consistent with that observed in [17] on several datasets. Nevertheless, the improvement achieved by using parallel L^3-nets (last row) suggests that each subnetwork yields an audio representation that is somewhat complementary to that of the other.

In summary, the proposed method (underlined in Table II) appears to be the best performing in terms of classification accuracy (89.01%). However, if limited computational resources are available, one may as well consider using a single L^3-net along with the avg ‖ max aggregation strategy. Indeed, the two 4.68 M-parameter models individually achieve competitive performances, with a classification accuracy of 86.51% and 87.28% for the *env* and *music* L^3-nets, respectively, outperforming [12] by approximately 5% despite having more than 17 times fewer parameters.

VI. CONCLUSIONS

In this manuscript, we presented ITTV, a freely-available dataset of Italian TV programs gathered from YouTube and manually annotated by genre. Furthermore, we described a novel method for audio-based automatic genre classification of media broadcasts based on L^3-net embeddings that achieves state-of-the-art results. Finally, by showing that end-to-end neural networks equipped with late majority voting fail to discriminate acoustically-similar TV genres, we empirically substantiated the choice of an early aggregation stage in which statistical modeling is performed across the entire program prior to the downstream classifier. Future work includes combining existing methods based on collections of sound events with the proposed feature representations to benefit from the advantages of the two approaches.

REFERENCES

[1] Cees GM Snoek, Marcel Worring, and Arnold WM Smeulders, "Early versus late fusion in semantic video analysis," in *Proc. 13th Annu. ACM Int. Conf. Multimedia*, 2005, pp. 399–402.

[2] Stéphane Ayache, Georges Quénot, and Jérôme Gensel, "Classifier fusion for SVM-based multimedia semantic indexing," in *Eur. Conf. Inf. Retrieval*, 2007, pp. 494–504.

[3] Maurizio Montagnuolo and Alberto Messina, "Automatic genre classification of TV programmes using Gaussian mixture models and neural networks," in *Proc. 18th Int. Workshop Database Expert Syst. Appl.*, 2007, pp. 99–103.

[4] Maurizio Montagnuolo and Alberto Messina, "Parallel neural networks for multimodal video genre classification," *Multimedia Tools and Appl.*, vol. 41, no. 1, pp. 125–159, 2009.

[5] Yu-Gang Jiang, Xiaohong Zeng, Guangnan Ye, Dan Ellis, Shih-Fu Chang, Subhabrata Bhattacharya, and Mubarak Shah, "Columbia-UCF TRECVID2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching," in *TRECVID 2010 Workshop Participants Notebook Papers*, 2010, vol. 2, pp. 3–2.

[6] Zhen-zhong Lan, Lei Bao, Shoou-I Yu, Wei Liu, and Alexander G Hauptmann, "Multimedia classification and event detection using double fusion," *Multimedia Tools and Appl.*, vol. 71, no. 1, pp. 333–347, 2014.

[7] Mortaza Doulaty, Oscar Saz, Raymond W. M. Ng, and Thomas Hain, "Automatic genre and show identification of broadcast media," in *Proc. 17th Annu. Conf. Int. Speech Communication Association*, 2016, pp. 2115–2119.

[8] Sher Muhammad Daudpota, Atta Muhammad, and Junaid Baber, "Video genre identification using clustering-based shot detection algorithm," *Signal Image Video Process.*, vol. 13, no. 7, pp. 1413–1420, 2019.

[9] Hazim Kemal Ekenel, Tomas Semela, and Rainer Stiefelhagen, "Content-based video genre classification using multiple cues," in *Proc. 3rd Int. Workshop Automated Inf. Extraction Media Production*, 2010, pp. 21–26.

[10] Mortaza Doulaty, Oscar Saz, Raymond WM Ng, and Thomas Hain, "Latent Dirichlet allocation based organisation of broadcast media archives for deep neural network adaptation," in *Proc. 2015 IEEE Workshop Autom. Speech Recog. Understanding*, 2015, pp. 130–136.

[11] Samuel Kim, Panayiotis Georgiou, and Shrikanth Narayanan, "On-line genre classification of TV programs using audio content," in *Proc. 2013 IEEE Int. Conf. Acoust. Speech Signal Process.*, 2013, pp. 798–802.

[12] Lam Pham, Chris Baume, Qiuqiang Kong, Tassadaq Hussain, Wenwu Wang, and Mark Plumbley, "An audio-based deep learning framework for BBC television programme classification," in *Proc. 29th Eur. Signal Process. Conf.*, 2021, pp. 56–60.

[13] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2880–2894, 2020.

[14] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *Proc. 2017 IEEE Int. Conf. Acoust. Speech Signal Process.*, 2017, pp. 776–780.

[15] Relja Arandjelovic and Andrew Zisserman, "Look, listen and learn," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 609–617.

[16] Relja Arandjelovic and Andrew Zisserman, "Objects that sound," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 435–451.

[17] Jason Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *Proc. 2019 IEEE Int. Conf. Acoust. Speech Signal Process.*, 2019, pp. 3852–3856.

[18] Peter Bell, Mark J. F. Gales, Thomas Hain, Jonathan Kilgour, Pierre Lanchantin, Xunying Liu, Andrew McParland, Steve Renals, Oscar Saz, Mirjam Wester, and Philip C. Woodland, "The MGB challenge: Evaluating multi-genre broadcast media recognition," in *Proc. 2015 IEEE Workshop Autom. Speech Recog. Understanding*, 2015, pp. 687–693.

[19] Mingxing Tan and Quoc Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[20] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone, *Classification and regression trees*, Routledge, London, United Kingdom, 1984.