CLASSIFYING NON-SPEECH VOCALS: DEEP VS SIGNAL PROCESSING REPRESENTATIONS

Fatemeh Pishdadian, Bongjun Kim, Prem Seetharaman, Bryan Pardo

Northwestern University
Electrical Engineering and Computer Science
Evanston, IL, USA
{fpishdadian, bongjun, prem}@u.northwestern.edu, pardo@northwestern.edu

ABSTRACT

Deep-learning-based audio processing algorithms have become very popular over the past decade. Due to promising results reported for deep-learning-based methods on many tasks, some now argue that signal processing audio representations (e.g. magnitude spectrograms) should be entirely discarded, in favor of learning representations from data using deep networks. In this paper, we compare the effectiveness of representations output by state-of-the-art deep nets trained for task-specific problems, to off-the-shelf signal processing representations applied to those same tasks. We address two tasks: query by vocal imitation and singing technique classification. For query by vocal imitation, experimental results showed deep representations were dominated by signal-processing representations. For singing technique classification, neither approach was clearly dominant. These results indicate it would be premature to abandon traditional signal processing in favor of exclusively using deep networks.

Index Terms— audio signal representation, audio processing, audio classification, query by example, deep learning

1. INTRODUCTION

Recently, deep-learning-based audio processing has gained great popularity, due to the promising results these methods have produced for tasks such as audio classification [1] and audio source separation [2]. As a result, some argue that representations built using signal-processing knowledge and theory (e.g. Fourier transforms, cepstrograms, etc.) should be entirely discarded, in favor of learning representations from data using deep networks [3].

In this work, we study the efficacy of both deep and signal-processing representations in the context of content-based audio retrieval and audio classification, focusing on two example tasks within these broad categories: query by vocal imitation and singing technique classification. Given a collection of audio files, Query by vocal imitation (QBV) [4, 5, 6] aims to retrieve those files most similar to a user's vocal imitation of a sound (e.g. an imitation of dog barking). QBV is particularly useful when detailed text labels for audio samples are not available. Singing technique classification (e.g. Broadway belting, vocal fry) is useful for automated music instruction, genre recognition [7], and singer identification [8]. On both of these tasks, the current state-of-the-art reported in the literature uses a deep model to encode the audio.

This work was supported, in part, by USA NSF award 1617497.

We compare the effectiveness of representing the audio using a state-of-the-art deep model, trained specifically for a task, to the effectiveness of using one of three off-the-shelf signal processing representations. We use a nearest-neighbor classification framework to perform query by vocal imitation and singing technique classification. Audio queries and sound files in the database are both encoded in the same way (with either a deep net or a signal processing method), and then a nearest-neighbor classification is performed to find the database example most like the query. If deep representations are truly better for these tasks, then encoding audio with a task-specific deep model should make the task-salient information more prominent than encoding with a signal processing method. This should translate to better performance.

The results of this study are not what recent literature would lead one to expect. The representation that stands out as the most useful is the 2D Fourier transform of a constant-Q spectrogram, rather than the representation produced by any deep network. These results indicate it would be premature to abandon traditional signal processing approaches in favor of exclusively using deep networks.

2. DEEP REPRESENTATIONS

In query by vocalization (QBV), people tend to remain more faithful to the general shape of spectral modulations rather than the exact pitch or timing of a reference audio. Therefore, a QBV representation should be able to capture modulation patterns and also be robust against small deviations in the pitch or timing of a query with respect to the target sound. Similarly, singing techniques create modulation patterns that serve as powerful discriminants for singing styles. It is thus desirable for an audio representation used for these tasks to preserve the modulations and present them explicitly.

Convolutional layers in deep networks are known to be effective in capturing shift-invariant patterns. For instance, if the input to a convolutional layer is an audio spectorgram, the layer can be trained to extract up-/downward moving spectral patterns, i.e. spectro-temporal modulations, regardless of their start time and offset frequency [9]. It is, therefore, not surprising that convolutional nets (CNNs) are the current state-of-the-art on both tasks. We now describe the specific networks used in our experiments.

TL-IMINET [10] is a deep net built specifically for query by vocal imitation (QBV). The trained network takes a pair of audio recordings as input: a vocal imitation (e.g. a human imitation of a dog bark) and an original recording (e.g. a real dog bark) and outputs a similarity rating ranging from 0 to 1. TL-IMINET has two convolutional towers that feed into several fully connected layers that combine input from the two towers. Each tower has three

https://doi.org/10.33682/bczf-wv12

convolutional layers with max-pooling. The specific filters (their number, arrangement, etc.) differ between the two towers, as one was designed to capture features from an original recording and the other from a vocal imitation. The authors argue that the convolutional towers capture spectro-temporal modulation patterns resembling feature maps used by mammals in the auditory system. Each CNN tower takes a 4 second-long log-mel spectrogram as input. We use a replication of TL-IMINET, trained on the same data sets to a performance level equal to that reported in the original paper. One can consider this network a specialist for the QBV task.

The *VGGish model* [11] is a CNN-based model trained on the audio from 8 million YouTube videos to distinguish 3,000 sound classes. It has 6 convolutional layers, followed by 3 fully-connected layers. It takes a log-mel spectrogram (64 Mel bins, window size of 25 ms, and hop size of 10 ms) as input and outputs a 128-dimensional feature embedding for every 1-second segment of the input audio. We selected it as an example of deep network architectures and trained models used for a variety of audio labeling tasks. One can consider the audio embedding produced by VGGish as a "general" audio representation. As such, it is used as a sanity-check baseline for both QBV and singing technique classification. No task-specific model should do worse than this general audio model.

Modified VGGish [12] (M-VGGish) is a network for query by vocalization. Instead of extracting the feature embedding from the final layer of a VGGish model, the authors used intermediate representations from the convolutional layers, which resulted in better QBV performance than the original VGGish feature embedding. The model takes an arbitrary length recording and outputs a feature vector for every 2-second segment of the recording. To form the segment-level feature vector, the outputs from the last two convolutional layers are concatenated, then the set of segment-level feature vectors are averaged to form a clip-level feature vector. One can consider this network a specialist for the QBV task.

Wilkins et al. [13] made a convolutional neural network that is the current state-of-the-art for singing technique classification. It is an end-to-end model that takes a raw PCM audio waveform as input and outputs a probability distribution over singing techniques. It is a specialist for singing technique classification.

2.1. Signal-processing-based representations

In this section, we discuss the signal processing representations used in our experiments. Time-frequency representations, such as the magnitude spectrogram are, perhaps, the most commonly used audio features. For this study, we used a log-frequency magnitude spectrogram built using a *Constant-Q Transform* (CQT) [14]. The log-scale frequency spacing of the CQT preserves the spacing between overtones of harmonic sounds (e.g. human speech) when the fundamental frequency changes. A log-frequency magnitude spectrogram is used as input to three of the four deep models included in this study (TL-IMINET, VGGish, modified VGGish). Therefore, one can consider the CQT spectrogram a baseline. If a nearest-neighbor classifier performs better using a CQT spectrogram as input than it does using the output of one of these deep models, then that model is not performing task-relevant work.

The *2D Fourier Transform (2DFT)* is an image processing tool that was not originally developed for audio. It decomposes an image into a set of scaled and phase-shifted 2D sinusoids. The 2DFT can be used to analyze the time-frequency representations (e.g. CQT) of audio signals [15, 16]. Repeating patterns in a time-frequency representation, such as overtones of a harmonic sound, are grouped together and manifest as peaks in the 2DFT domain. Spectro-temporal modulation patterns can thus be effectively encoded by the 2DFT as a set of peaks. The magnitude 2DFT of an audio spectrogram is invariant with respect to frequency or time shifts of modulation patterns. The 2DFT has been recently used in applications such as music/voice separation [15] and cover song identification [17, 16]. Since it has proven successful in these very different tasks, we were interested in exploring its potential for the tasks in this study. In our experiments, we apply the 2DFT to the log-frequency magnitude spectrogram built from the CQT.

Scale-rate (*SR* in this work) is a modulation-related feature representation computed based on the *Multi-resolution Common Fate Transform (MCFT)* [18][1]. The MCFT is a bio-inspired representation initially proposed for the task of audio source separation, which encodes spectro-temporal modulation patterns as explicit dimensions. The modulation-related dimensions are termed *scale* and *rate* [19], respectively encoding the spectral spread and modulation velocity over time. SR is built by applying the 2D filterbank of the MCFT to the magnitude CQT of audio signals and averaging the results over time and frequency. This representation was chosen due to its explicit representation of spectro-temporal modulations, which we believe to be useful in both QBV and singing technique classification tasks.

While we had reason to believe that the 2DFT of the log spectrogram and the scale-rate representation would capture vocal spectro-temporal modulations well, none of the signal processing representations in our study were designed for either the QBV or singing technique classification task. This contrasts with the deep networks we tested, which were made specifically for each task.

3. EXPERIMENTS

We consider an audio representation (e.g. the deep embedding output by M-VGGish, or the CQT spectrogram) as effective if the task-relevant distinctions between audio examples can be easily captured by a similarity measure applied to those examples encoded in the representation. The performance of a K-nearest-neighbor classifier, for instance, is directly affected by the audio representation. The better the task-related information is represented, the better the classifier works, the better the retrieval (or labeling) performance would be. We evaluate the effectiveness of a representation in this light.

3.1. Query by vocal imitation

To evaluate the performance of the representations in the query by vocal imitation (QBV) scenario we used the VimSketch dataset[2], which combines the datasets from two previous publications [20, 21] to create the largest single dataset for QBV. VimSketch contains 542 reference sounds (including a variety of animal sounds, musical snippets, and environmental noise samples) and 12,543 vocal imitations of those reference sounds with a minimum of 13 and a maximum of 37 vocal imitations per reference.

All audio examples encoded by the representations were zero-padded to the length of the longest example in the dataset (15.4 seconds). All signal processing representations, VGGish and M-VGGish used full length audio examples. Audio examples for TL-IMINET were limited to the initial 4 seconds long. This duplicates the published approach for TL-IMINET [10].

[1] https://interactiveaudiolab.github.io/MCFT/

[2] http://doi.org/10.5281/zenodo.2596911

The VimSketch audio files were originally sampled at different rates (ranging from 8 kHz to 192 kHz). Thus, we first resampled them to have a common rate, 8 kHz. For the signal processing representations (CQT, 2DFT, and SR), the range of frequencies was limited to 55 Hz to 2.09 kHz (pitches A1 to C7). This was done to keep the computations tractable. Audio was upsampled to 16kHz to accommodate the input requirements of VGGish and M-VGGish. Input to TL-IMINET was upsampled to 16kHz for the imitation tower and 44.1kHz for the reference tower, to meet its requirements. We note that even though the initial resampling removes the frequencies above 4 kHz, the comparison is still reasonable since: i) these high frequencies are unavailable to all representations alike and ii) the sounds remain recognizable by humans.

Given a vocal query, the output of a QBV system is a list of reference sound examples ordered based on their similarity to the query. In our experiments with the deep net encoders (VGGish, M-VGGish) and all signal processing features (CQT, 2DFT, SR), we use the cosine similarity measure to select the most similar sound examples to a query in the representation domain. We selected the cosine similarity measure because this is the similarity measure applied in the published results for the state-of-the-art M-VGGish network for query by vocal imitation [21].

The cosine similarity between a query $\mathbf{V_q}$ and a reference $\mathbf{V_r}$ is defined as:

$$S_{cos}(\mathbf{V_q}, \mathbf{V_r}) = \frac{\langle \mathbf{V_q}, \mathbf{V_r} \rangle}{\|\mathbf{V_q}\| \|\mathbf{V_r}\|}, \quad (1)$$

where $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$ are the inner product and Euclidean norm.

Unlike VGGish and M-VGGish, TL-IMINET was not designed to produce an audio embedding (a.k.a. representation) to be used by an external similarity measure or classifier. It instead directly outputs a similarity measure between pairs of examples, which we use in place of the cosine similarity applied to all other representations.

The performance of the QBV system can be evaluated in terms of the rank of the target sound in the output list of sound examples. We use Mean Reciprocal Rank (MRR) as the performance measure:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}, \quad (2)$$

where Q denotes a set of queries and $rank_i$ refers to the rank position of the target sound for the ith query.

Recall that a 'reference' in this context is an audio file in the collection (e.g. a car horn recording) and a 'query' is a vocal imitation of some reference file. The MRR value for each representation is computed over datasets of size n = 20, 50, 100, 200, 400, and 542 references. Since there are 12,543 vocal imitations in the data, the MRR value for each reference set is computed by averaging over 12,543 reciprocal ranks. To ensure no result is due to a selection of a reference set that is skewed to favor a particular representation, reference set selection is repeated 100 times for each value of n below 542 (the size of the full reference set). The average MRR over all 100 iterations is reported.

3.1.1. Signal Processing Hyperparameters

In computing the CQT and 2DFT, we treat the frequency resolution of the CQT as a tunable parameter, taking on values of 12, 24, 48, or 96 bins/octave. For SR, we keep the frequency resolution fixed to the best performing 2DFT resolution for the QBV task and then treat the scale and rate resolutions as tunable parameters with values 1, 2, 4, or 8 bins/octave.

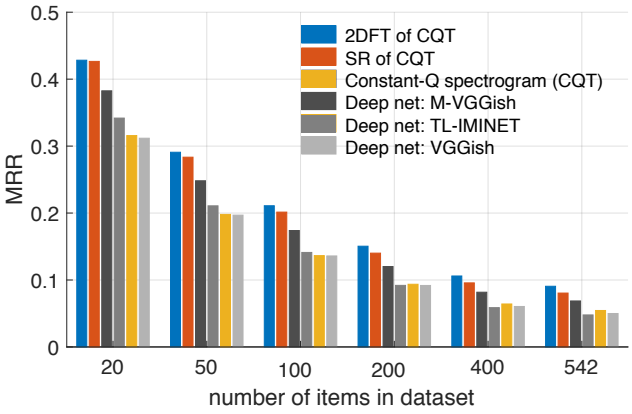

Figure 1: Query by vocalization search results. Deep representations (grayscale bars) are compared to signal processing representations (colored bars). Higher values are better. Each signal processing representation uses the **worst** parameters found for the task. Two of the deep nets (M-VGGish and TL-IMINET) were constructed specifically for query by vocalization. Nevertheless, all deep representations are dominated by two signal processing representations (2DFT of CQT and SR).

For the QBV task, the worst and best results using the CQT and 2DFT are obtained with a frequency resolution of 96 (worst) and 12 bins/octave (best). Increasing the frequency resolution has a negative effect on the performance of both features, showing the importance of high temporal resolutions in capturing the fine structure of modulations. The scale resolution does not impact the performance of SR features significantly, and hence it is fixed to 1 bin/octave for the reported results. The rate resolution, on the other hand, has a noticeable effect, giving the worst results when set to 1 bin/octave and best results when set to 8 bins/octave.

3.1.2. Results

QBV task results are presented in Figure 1. To tilt the comparison in favor of the deep representations as far as possible, we show only the results for the **worst** tunable parameter settings found for the signal processing approaches (CQT, 2DFT, and SR). Results shown for 2DFT and CQT use the **worst** frequency resolution tested. Results for SR show the **worst** scale and rate tested. Therefore, Figure 1 compares off-the-shelf signal processing representations that use bad hyperparameter choices to published task-specific deep models, tuned to work well on the kind of data used for evaluation.

It can be clearly observed that the 2DFT and SR features outperformed all other representations, even with their worst parameter selection. This superiority holds for all sizes of dataset tested. CQT is a log-frequency spectrogram. VGGish, M-VGGish, and TL-IMINET all use a log-frequency spectrogram as input. VGGish was neither trained nor designed for the QBV task and serves as a baseline among the deep nets. Therefore, it is not surprising that using the output of VGGish as a representation is roughly equivalent to simply using a constant-Q spectrogram (the CQT).

The deep nets TL-IMINET and M-VGGish were both constructed for the specific QBV task we tested them on. Surprisingly, TL-IMINET, a network designed and trained specifically for the QBV task, shows degraded performance as the dataset grows, to the point where it is would actually be preferable to use a constant-Q spectrogram, which was the worst-performing of the signal process-

ing representations.

As can be seen, only M-VGGish consistently improved upon CQT as the database size increases. This shows it is possible to create a deep representation that is consistently better than its input representation for this task. That said, M-VGGish never achieved the performance of either of the top signal processing representations (2DFT and SR), despite the fact that we compared to the worst parameter settings for both. We hypothesize that the superior performance of the signal processing representations on this task may be due to the fact that SR and 2DFT features inherently capture spectro-temporal modulations and present them in a time/frequency-shift-invariant fashion. While a deep representation may be able to represent such modulations, this comparison illustrates that even a network explicitly designed and trained for this task (e.g. TL-IMINET) may not perform as well as an existing signal processing approach.

3.2. Singing technique classification

For the task of singing technique classification, we used VocalSet [13], a singing voice dataset that includes a large set of voices (9 female and 11 male professional singers) performing 17 different singing techniques. We extracted the samples corresponding to 10 different singing techniques (belt, breathy, inhaled singing, lip trill, spoken excerpt, straight tone, trill, trillo, vibrato, and vocal fry) by all singers, which amounts to 915 samples ranging in length from 1.7 to 21.5 seconds. All audio examples were resampled to 8 kHz.

We compared our results to those of the classifier proposed by Wilkins et al. [13], which was directly trained on VocalSet, and thus is expected to perform very well. Their classifier is a neural network, composed of three convolutional layers followed by two dense layers. The network receives a 3-second time-domain audio excerpt as input and outputs the predicted vocal technique class. We use the same technique classes and the same training/testing data split as in their experiments. The training and testing sets include samples from 15 and 5 singers, respectively.

The signal processing representations compared to Wilkins et al. were CQT, 2DFT, and SR. We also compared to VGGish, a deep net not trained on this specific task. This provided a baseline deep net, much the way the CQT provides a baseline signal processing representation. The singing techniques were classified using the K-Nearest Neighbors algorithm, with the cosine similarity measure and $K = 3$ used as algorithm parameters in all experiments.

Since audio examples are of different lengths, we had to decide whether to zero-pad or cut all files to the same length. Two lengths were tried. First, we extracted the initial 3 seconds of all examples, which is the same length used by the deep net in Wilkins et al. [13]. We expected this to favor their deep net. Next, we found the signal length that maximized the performance of the VGGish deep net (18 seconds) and zero-padded or cut all examples to that length.

We measured the classification performance in terms of precision, recall, and F-measure. Table 1 shows results for 3-second examples and Table 2 the results for 18-second examples.

The frequency range for the CQT and the parameter tuning strategies for the 2DFT and SR features were the same as in Section 3.1. Since the signal processing approaches were not as dominant in this task, we report the results using both the best and the worst parameter settings for these representations. In both tables, the best frequency resolutions for the CQT and 2DFT are 96 and 24 bins/octave, respectively. In both tables, the best scale resolution is 1 bin/octave and the best rate resolution 8 bins/octave.

Representation	Precision	Recall	F-measure
Deep: Wilkins et al.	**0.677**	**0.628**	**0.651**
Deep: VGGish	0.556	0.54	0.529
CQT-best	0.61	0.528	0.519
CQT-worst	0.52	0.468	0.448
2DFT-best	**0.665**	**0.624**	**0.637**
2DFT-worst	**0.660**	**0.58**	**0.597**
SR-best	0.562	0.564	0.554
SR-worst	0.449	0.44	0.434

Table 1: Singing technique classification (10 classes): Results for 3-second excerpts. Higher values are better.

Representation	Precision	Recall	F-measure
Deep: VGGish	0.627	0.6	0.602
CQT-best	0.533	0.488	0.479
CQT-worst	0.43	0.432	0.408
2DFT-best	**0.723**	**0.692**	**0.698**
2DFT-worst	**0.674**	**0.636**	**0.646**
SR-best	0.615	0.612	0.603
SR-worst	0.612	0.6	0.599

Table 2: Singing technique classification (10 classes): Results for 18-second excerpts. Higher values are better.

It can be observed that in the 3-second case, the 2DFT outperforms the VGGish embeddings by a large margin and a simple parameter tuning (frequency resolution) brings its performance close to the network that was specifically trained for the VocalSet data (Wilkins et al.). When applied to excerpts of longer duration (Table 2), the 2DFT is able to capture long-term modulations even more efficiently, yielding a higher F-measure than the state-of-the-art results reported by Wilkins et al. on 3-second examples.

4. CONCLUSION

For query by vocalization, a nearest-neighbor method that applies cosine similarity to either of two off-the-shelf signal processing methods (2DFT and SR applied to a constant-Q spectrogram) outperformed similarity measures built using two different deep approaches designed specifically for this task (M-VGGish and TL-IMINET), as well as a general audio deep representation (VGGish). For singer technique classification, a 2DFT representation was competitive with or outperformed the task-specific deep network that is the current state-of-the-art (Wilkins et al. [13]), depending on the choice of parameters, and also outperformed a general audio representation (VGGish).

The deep networks evaluated here defined the state of the art on both tasks until this study. We hypothesize that the ability of both SR and 2DFT to explicitly represent spectro-temporal modulations in a time/frequency-shift-invariant fashion is key to their effectiveness with non-speech vocal classification. While a deep representation may be able to represent such modulations, this comparison illustrates that even a network explicitly designed and trained for non-speech vocal classification may not perform as well at representing these features. Given our results, it would be premature to abandon traditional signal processing techniques in favor of exclusively using deep networks.

5. REFERENCES

[1] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.

[2] Y. Luo, Z. Chen, J. R. Hershey, J. Le Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: Stronger together," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 61–65.

[3] S. Venkataramani and P. Smaragdis, "End-to-end networks for supervised single-channel speech separation," *arXiv preprint arXiv:1810.02568*, 2018.

[4] M. Cartwright and B. Pardo, "Synthassist: an audio synthesizer programmed with vocal imitation," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 741–742.

[5] G. Lemaitre, O. Houix, F. Voisin, N. Misdariis, and P. Susini, "Vocal imitations of non-vocal sounds," *PloS one*, vol. 11, no. 12, p. e0168167, 2016.

[6] A. Mehrabi, K. Choi, S. Dixon, and M. Sandler, "Similarity measures for vocal-based drum sample retrieval using deep convolutional auto-encoders," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 356–360.

[7] W.-H. Tsai, D. Rodgers, and H.-M. Wang, "Blind clustering of popular music recordings based on singer voice characteristics," *Computer Music Journal*, vol. 28, no. 3, pp. 68–78, 2004.

[8] M. A. Bartsch and G. H. Wakefield, "Singing voice identification using spectral envelope estimation," *IEEE Transactions on speech and audio processing*, vol. 12, no. 2, pp. 100–109, 2004.

[9] J. Schlüter, "Learning to pinpoint singing voice from weakly labeled examples." in *ISMIR*, 2016, pp. 44–50.

[10] Y. Zhang, B. Pardo, and Z. Duan, "Siamese style convolutional neural networks for sound search by vocal imitation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 429–441, 2019.

[11] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, "Cnn architectures for large-scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2017, pp. 131–135.

[12] B. Kim and B. Pardo, "Improving content-based audio retrieval by vocal imitation feedback," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 4100–4104.

[13] J. Wilkins, P. Seetharaman, A. Wahl, and B. Pardo, "Vocalset: A singing voice dataset," in *Proc. 19th Conf. Int. Society for Music Information Retrieval (ISMIR)*, 2018, pp. 468–474.

[14] C. Schörkhuber, A. Klapuri, N. Holighaus, and M. Dörfler, "A matlab toolbox for efficient perfect reconstruction time-frequency transforms with log-frequency resolution," in *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[15] P. Seetharaman, F. Pishdadian, and B. Pardo, "Music/voice separation using the 2d fourier transform," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 36–40.

[16] D. P. Ellis and B.-M. Thierry, "Large-scale cover song recognition using the 2d fourier transform magnitude," 2012.

[17] P. Seetharaman and Z. Rafii, "Cover song identification with 2d fourier transform sequences," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 616–620.

[18] F. Pishdadian and B. Pardo, "Multi-resolution common fate transform," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 342–354, 2019.

[19] T. Chi, P. Ru, and S. A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 887–906, 2005.

[20] M. Cartwright and B. Pardo, "Vocalsketch: Vocally imitating audio concepts," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 43–46.

[21] B. Kim, M. Ghei, B. Pardo, and Z. Duan, "Vocal imitation set: a dataset of vocally imitated sound events using the audioset ontology," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2018.