CONTENT-BASED REPRESENTATIONS OF AUDIO USING SIAMESE NEURAL NETWORKS

Pranay Manocha [†◇], *Rohan Badlani* [⋆◇], *Anurag Kumar* [‡], *Ankit Shah* [‡], *Benjamin Elizalde* [‡§], *Bhiksha Raj* [‡]

[†]Department of Electronics and Electrical Engineering IIT Guwahati, India
[⋆]Department of Computer Science, BITS Pilani, India
[‡]Language Technologies Institute, Carnegie Mellon University, Pittsburgh, United States

Email: pranaymnch@gmail.com, rohan.badlani@gmail.com, alnu@andrew.cmu.edu, aps1@andrew.cmu.edu, bmartin1@andrew.cmu.edu, bhiksha@cs.cmu.edu

ABSTRACT

In this paper, we focus on the problem of content-based retrieval for audio, which aims to retrieve all semantically similar audio recordings for a given audio clip query. This problem is similar to the problem of query by example of audio, which aims to retrieve media samples from a database, which are similar to the user-provided example. We propose a novel approach which encodes the audio into a vector representation using Siamese Neural Networks. The goal is to obtain an encoding similar for files belonging to the same audio class, thus allowing retrieval of semantically similar audio. Using simple similarity measures such as those based on simple euclidean distance and cosine similarity we show that these representations can be very effectively used for retrieving recordings similar in audio content.

Index Terms— Audio Fingerprinting, Content-Based Retrieval, Query by Example, Siamese Network, Similar Matching

1. INTRODUCTION

Humans have an inherent ability to distinguish and recognize different sounds. Moreover, we are also able to relate and match similar sounds. In fact, we have the capability to detect and relate sound events or "acoustic objects" which we have never encountered before, based on how that phenomenon stands out against the background [1]. This ability plays a crucial role in our interactions with the surroundings and it is also expected that machines have this ability to relate the two audio recordings based on their semantic content. This is precisely the broad goal of this paper. We propose a method to encode semantic content of an audio recording such that two audio recordings of similar content (containing same audio events) can be matched and related through these embeddings. More specifically, we address the problem of content-based retrieval for audio: given an input audio recording we intend to retrieve audio recordings which are semantically related to it.

Semantic similarity matching and retrieval based on it has received much attention for video and images [2], [3] and [4]. However, in the broad field of machine learning for audio, semantic similarity matching and retrieval based on audio has received limited attention [5]. A major focus has been music information retrieval [6], [7], [8] and semantic retrieval of audio using text queries [9], [10]. Our focus here is on non-music and non-speech content, sounds which we hear everyday in our daily life, since they play an important role in defining the overall semantic content of an audio recording. Note that the problem of semantic content matching is different from the problem of audio event detection and classification [11]. Our goal is not to detect or classify sound events in an audio recording but to develop methods which capture the semantic content of an audio and be useful in retrieving similar audios. One method which has been explored considerably for audios is the idea of fingerprinting.

Audio fingerprinting is an acoustic approach that provides the ability to derive a compact representation which can be efficiently matched against other audio clips to compare their similarity or dissimilarity [12]. Audio fingerprinting has various applications like Broadcast Monitoring[13], Audio/Song Detection[14], Filtering Technology for File Sharing[15] and Automatic Music Library organization[16].

We focus on developing an efficient content-based retrieval system that can retrieve audio clips which contain similar audio events as the query audio clip. One can potentially think of applying the conventional fingerprinting approach [14] for matching to find recordings with similar audio content. However, fingerprinting is useful only in finding *exact match*. It has been used for finding multiple videos of the same event [17]. In [18] it is used to find multiple occurrences of a sound event in a recording. But it cannot solve the problem of retrieving all semantically similar files together. In fact even for finding repetitions of the same sound event it does not work well if the sound event is unstructured [18]. The reason is that fingerprinting tries to capture local features specific to an audio recording. It does not try to capture the broader features which might represent a semantically meaningful class. For searching similar videos based on content, we need audio retrievals belonging to the correct audio class and not just exact matches as in conventional fingerprinting. Hence, we need representations which can encode class specific information. In this paper, we try to achieve this by using a Siamese Neural Network. Although the siamese network has been previously explored for representations and content understanding in images and video [19, 3], to the best of our knowledge this is the first work employing it in the context of sound events.

Siamese neural networks incorporate methods that excel at detecting similar instances but fail to offer robust solutions that may be applied to other types of problems like classification. In this paper, we present a novel approach that uses a Siamese network to automatically acquire features which enable the model to distinguish between clips containing distinct audio events and encodes a given audio into a vector fingerprint. We show that the output feature vector has an inherent property to capture semantic similarity between audio containing same events. Although the cost of the learning al-

◇ First two authors contributed equally

§Acknowledges CONACYT for his doctoral fellowship, No.343964

gorithm itself may be considerable, this compressed representation is powerful as we are able to not only learn them without imposing strong priors like in [14], but also to retrieve semantically similar clips by using this feature space.

2. PROPOSED APPROACH

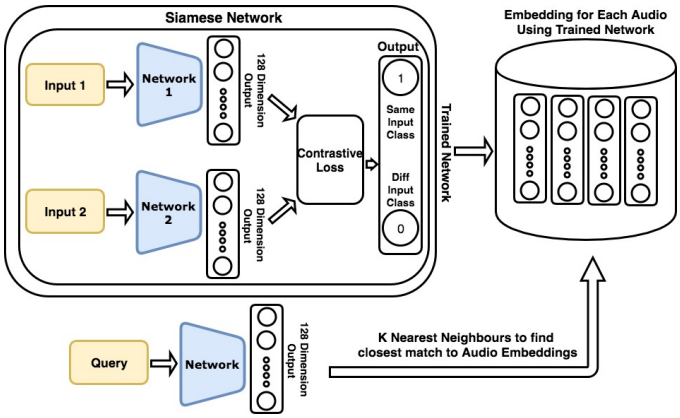

Fig. 1. Framework of the Proposed approach

2.1. Framework Outline

We propose a neural network based approach to obtain representations or embeddings such that semantically similar audios have similar embeddings.

We learn these semantic representations through Siamese Neural Networks. Fig 1 shows our proposed framework. A Siamese neural network actually consists of two twin networks. The Siamese network takes in two different inputs, one applied to each twin, and is trained to learn the similarity between these inputs. If the inputs are similar then it should predict 1 otherwise 0. We use the trained component (twin) network as a feature extractor to obtain representations for audio recordings in the database, as shown in the figure. The input audio query is also embedded through the same network and its embedding is matched with embeddings of recordings in the database to rank them in decreasing order of similarity. This ranking can be done through any distance or similarity measure. In this work we use cosine similarity and euclidean distance. Based on the ranked list one can return the *top K* most similar audios.

2.2. Siamese Network and Loss Function

The Siamese neural network is a class of neural network architectures that contains two or more identical sub-networks, meaning that all sub-networks have the same configuration with the same parameters. Weights and the parameter updates are mirrored across all sub-networks simultaneously. Siamese networks have previously been used in tasks involving similarity or identifying relationships between two or more comparable things. Muller et al.[20] used a Siamese network for paraphrase scoring by giving a score to a pair of input sentences. Bromley et al.[21] used a Siamese network for the task of signature verification. In the domain of audio, it has been incorporated for content-based matching in music [22] and in speech to model speaker related information [23, 24].

Siamese networks offers several advantages. All subnetworks have similar weights which leads to fewer training parameters thus requiring less training data and a lesser tendency to over fit. Moreover, the outputs of each of the subnetworks are representation vectors with the same semantics and this makes them much easier to compare with one other. These characteristics makes them well suited for our task.

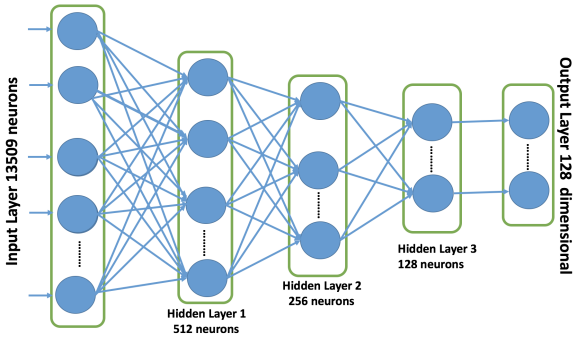

Fig. 2. Architecture of the Subnetworks in the Siamese Network. The final layer of 128 neurons is also the output layer

To train our Siamese Network we use the contrastive loss function defined in [25]. The goal is to learn the parameters W of a function G_W, such that neighbors are pulled together and non-neighbors are pushed apart. To achieve this, the objective function needs to be suitably defined. The learning process here operates on a pair of samples. Let X_1 , $X_2 \in \mathcal{P}$ be a pair of input samples and let Y be the label assigned to this pair. $Y = 1$ if the inputs X_1 and X_2 are similar, otherwise $Y = 0$. The distance between X_1 and X_2 is defined as the euclidean distance between the mapping from the function G_W

$$D_W(X_1, X_2) = \| G_w(X_1) - G_w(X_2) \| \quad (1)$$

The overall loss function for the pair X_1 and X_2, is then defined as

$$L(W, Y, X_1, X_2) = (Y)\frac{1}{2}(D_w)^2 + (1 - Y)\frac{1}{2}\{max(0, m - D_w)\}^2 \quad (2)$$

In Eq 2, $m > 0$ is the margin. The idea behind this margin is that the dissimilar points contribute to the training loss only if the distance between them, D_W, is within the radius defined by margin value m. For the pairs of similar inputs we always want reduce the distance between them.

2.3. Network Architecture

The architecture of the individual sub-networks in the Siamese network is shown in Fig 2. Each sub-network is a feed-forward multi layer perceptron (MLP) network. The input to the network are log-frequency spectrograms of audio recordings. The frames in Logspec are concatenated to create one long extended vector. The dimensionality of the inputs are 13509 (See 3.2 for details). The network consists of a total of 3 layers after the input layer. The first layer consists of 512 neurons, the second layer 256 neurons and the last layer has 128 neurons. The last layer also serves as the output layer. The activation function in all layers is ReLU ($max(0, x)$). A dropout of 0.3 is applied between all three layers during training. We will refer to the network as $\mathcal{N}_R$

2.4. Representations and Retrieval

All audio clips in the audio database are represented through the 128 dimensional output from the network $\mathcal{N}_R$. When a query audio clip is given, we first obtain its 128 dimensional representation using $\mathcal{N}_R$. This representation is then matched to representations of all audios in the database using a similarity or distance function. The clips in database are ranked according to the similarity measure and then the top K clips are returned. In other ways, one can think of it as obtaining K nearest neighbors in the database. Note that all

operations are done on fixed length audios of 2 seconds, details are provided in further sections.

3. DATASET AND EXPERIMENTAL SETUP

We study the problem of semantic similarity based retrieval in the domain of sound events. More specifically, given an audio clip of a sound class, the goal is to retrieve all audio clips of that class from the database. We consider the list of sound events from 3 databases, ESC-50[26], US8K[27] and TUT 2016[28]. Overall, we considered a total of 76 sound events. Some examples are, *Dog Barking, Clock Tick-Tock, Wind Blowing* etc. audio events include wide range of sound events, including different sound events from broad categories from animal sounds such as *Dog Barking and Crow*, non-speech human sounds such as *Clapping and Coughing*, exterior sounds such as *Siren, Engine, Airplane* to urban soundscape sounds such as *Street Music, Jackhammer* etc.

3.1. YouTube Dataset

The importance of semantic similarity matching lies in its utility in content-based retrieval of multimedia data on the web. Hence, we work with audio recordings from YouTube. User generated recordings on multimedia sharing websites such as YouTube, are often very noisy, unstructured and recorded under different conditions. Hence, even *intra-class* variation is very high, which makes content-based retrieval an extremely difficult problem.

For each of the 76 classes, we obtain 100 recordings from YouTube. To obtain relevant results we use <SOUND_NAME sound >(e.g <car horn sound >) as search query. From the returned list we select 100 recordings based on their length and relevance for the query. Very short (<2 seconds) and very long (>10 min) recordings are not considered.

We divided the dataset in the ratio of 70-10-20. 70 percent of the data per class is used for training and the remaining 30 percent data is split 1:2 between validation and testing. Thus, we take 70 samples per class for training, 10 for validation and the remaining 20 for testing, given that we roughly have 100 files per audio class. Overall, we have around 5K audio files for training, 760 files for validation and around 1500 files for testing. For our experiments, we operate on 2 second clips from each of these recordings. Hence, our actual database for experiments are fixed length 2 second audio clips in training as well as validation and test sets.

3.2. Siamese Network Training

The inputs to the Siamese network must be pairs of audio clips. We assign label 1 to the pairs of clips from the same class and label 0 to the pairs from different classes. We consider two training sets, balanced and unbalanced. The network trained on balanced set will be referred to as $\mathcal{N}_R^B$ and that trained on unbalanced as $\mathcal{N}_R^U$. In the balanced case, to create pairs with positive label ($Y = 1$), we consider all possible pairs belonging to the same audio class. For pairs with negative label ($Y = 0$), a clip belonging to a sound class is randomly paired with a clip from any other sound class. Hence, we end up with equal number of positive and negative label pairs. In the unbalanced case the positive label pairs are obtained in the same way. But for the negative label, we pair a clip belonging to a sound class with all clips not belonging to that sound. Thus, we have a non equal distribution of positive and negative labels.

We used the log spectrogram features, taking 1024 point FFT over a window size of 64ms and an overlap of 32ms per window. Both the axis were converted to the log scale and 79 bins were chosen for the frequency axis whereas 171 quantization bins were chosen for the time scale. We then concatenate these 79X171=13509 features and use as an input to the Siamese Network. The reason for taking log spectrogram features is that the features having a large difference on a common scale have diminished differences on the log scale. It is specially useful in those cases where we have a huge variation of feature magnitudes and hence it brings all a common scale. All parameters were tuned using the validation set. We train each model to 200 epochs and optimize on the training and validation losses.

3.3. Retrieval

We obtain the vector encoding of each file of the audio class by passing it through the trained saved model. All audio clips in the database are represented by these representations. At the time of testing, we obtain the representation for the query audio clip using the network and compute its similarity with representations of audios in the database. For computing similarity between two representations we use either euclidean distance or cosine similarity.

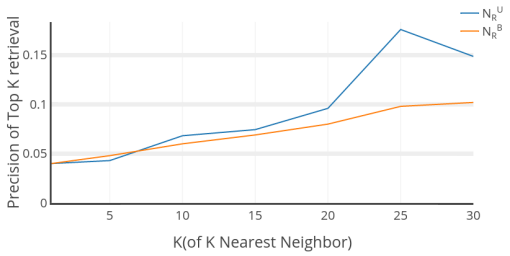

Fig. 3. Variation of MP^K with different K, from K=1 to K=30

4. EVALUATION AND RESULTS

4.1. Metrics

For any given query audio, we obtain a ranked list of audio clips present in the database which contain similar audio events present in the query clip. We then compute 3 metrics for evaluation which are defined below:

4.1.1. Average Precision

The average precision for a query is defined as mean of precisions at all positive hits.

$$AP = \frac{1}{m_j} \sum_{i=1}^{m_j} Precision_i \tag{3}$$

$Precision_i$ measures the fraction of correct items among first i recommendations. This precision is measured at every positive hit in the ranked list. m_j refers to the number of positive items in the database for the query. Average precision is simply the mean of these precision values. We will be reporting the mean of average precision (MAP) over all queries.

4.1.2. Precision at 1

This metric measures the precision at the first positive hit only. The idea is to understand where does the first positive item lie in the ranked list. Again the mean of Precision at 1 (MP^1) over all queries are reported.

4.1.3. Precision of Top K retrieval

This metric measures the quality of retrieved items in the top K items in the ranked list. For each query, we calculate the number of correct class instances in the top K files and then divide that by K to get the precision of the correct class amongst the top K retrieved files and take an average across all queries. Multiplying this score by K tells us the average number of correct class matches in the top K of the retrieved list. This measure tells us about the precision of the correct class in the top K retrieved list. Once again the mean of this metric over all queries is reported (MP^K)

Measures	$\mathcal{N}_R^B$	$\mathcal{N}_R^U$
MAP	0.0241	0.0342
MP^1	0.314	0.436
$MP^{K=25}$	0.099	0.177

Measures	$\mathcal{N}_R^B$	$\mathcal{N}_R^U$
MAP	0.0186	0.0133
MP^1	0.132	0.333
$MP^{K=25}$	0.105	0.133

Table 1. Left: Performance using euclidean distance, Right: Performance using cosine similarity

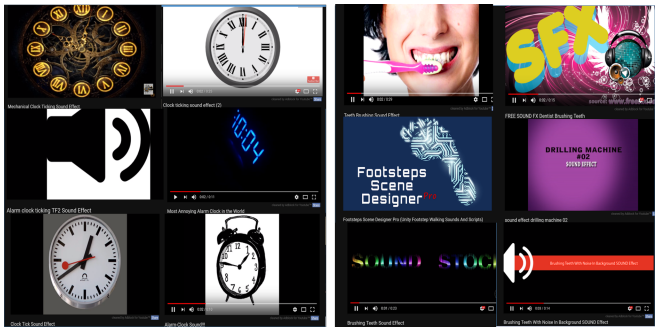

Fig. 4. Examples of content-based retrieval. Left: Clock Tick, Right: Brushing Teeth

The variation of MP^K with K is shown in Figure 3. We observe that this metric is maximum around K=25 and hence we report the best possible performance from now on for K=25.

4.2. Results and Discussion

We first show performance with respect to queries. From table 1, we observe that the Euclidean distance performance exceeds the Cosine similarity performance in the Mean Average Precision measure. This may be due to the fact during siamese network training, euclidean distance is used to measure the closeness between two points. Hence, the learned representations are inherently designed to work better with euclidean distance.

We note that the overall MAP of the system is similar to what has been traditionally observed throughout audio retrieval work [29]. MP^1 value of around 0.3 (for $\mathcal{N}_R^B$) indicates that the first positive hit on an average is achieved at rank 3. However, for a given specific query it can be much better. We note that the MP^1 values are fairly high, implying that the first positive hit can be easily obtained using the audio embeddings generated using Siamese Network. Also, note that the network $\mathcal{N}_R^U$ performs much better compared to $\mathcal{N}_R^B$. $\mathcal{N}_R^U$ is trained using a larger set of pairs of dissimilar audios and hence it is able learn more discriminitive representations.

The most important metric for understanding performance of a retrieval system is MP^K. The values for MP^K multiplied by K gives us the average number of correct class instances in the top K for a query. A low value of this measure means that a low number of correct class instances are obtained in the top K retrieved files. We are able to obtain fairly reasonable value of MP^K.

Fig4 gives visualization of a retrieval example. It shows two examples of queries and their top 6 retrieved similar files. We observe that for the class 'clock tick sound', the retrieval is from classes 'clock tick sound' and 'clock alarm sound', which are both nearly similar audio events. For the class 'brushing teeth sound', the system performs well as their are no other similar audio classes in the database and hence it retrieves 5 out of the 6 files correctly. Overall, it illustrates that our system is capable of delivering content based retrieval of audio recordings.

We now show performance on retrieval for some specific sound events. We show the average MP^K measure over all queries of a sound event class. Due to space constraints, we are not able to show performance numbers for all classes. In Table 2, we show performance for 20 sound events, 10 with highest MP^K values and 10 with lowest MP^K values. First, we note that for several classes we are able to obtain reasonably high performance using our learned representations. For example, for *Wind Blowing*, around 20 out of the top 25 retrieval actually belong to the sound class wind blowing. However, we also observe that some audio classes have unusually low average precisions. This occurs because we combine the labels of different datasets, there were some similar events classes which are semantically same but were treated as separate classes like 'Dog bark sound' and 'Dog sound' and 'Car horn sound' and 'Car passing by residential area'. Hence, the problem is more of how a sound event is referred to as instead of the actual representation we obtain from our method. We are actively investigating this problem of similar audio events with minor differences in text labels and will be addressing this in our future work.

Audio Class	$MP^{K=25}$
Wind Blowing	0.784
Sheep	0.753
Pig	0.724
Water Drop	0.711
Clock Tick	0.708
Brushing Teeth	0.699
Drilling Sound	0.681
Helicopter	0.670
Chirping Birds	0.650
Rooster	0.636

Audio Class	$MP^{K=25}$
Dog Bark	0.110
Car Horn	0.272
Crackling Fire	0.276
Glass Breaking	0.312
Can Opening	0.314
Crying Baby	0.315
Gun Shot	0.318
Crickets(insect)	0.41
Banging Residential Area	0.42
Children Playing	0.44

Table 2. Left: Classes (Top 10) with highest precisions, Right:Classes (bottom 10) with least precision

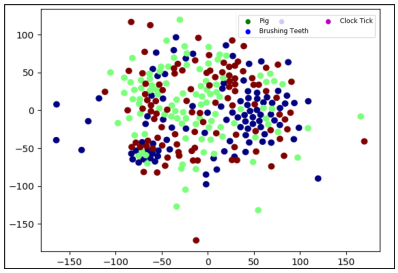

Fig. 5. TSNE plot for 3 classes- Clock Tick Sound, Brushing Teeth Sound, Pig Sound

Figure 5 shows that the visualization of 2 dimensional t-SNE embeddings for the 3 classes. For the purpose of clarity, we included only 3 classes in the plot. Once can see that the representations learned for audios are actually encoding semantic content as audios of same class cluster close to each other. This demonstrates that the proposed Siamese Network based representations inherently indistinguish between distinct audio events.

5. CONCLUSIONS

We proposed a novel approach that uses Siamese Neural Network to learn representations for audios. Our results indicate that these representations are able to capture semantic similarity between audio containing same events. This makes them well suited for content based retrieval of audio. We observe that for several classes, the precision of top 25 results is very high. We tried different measures of similarity like conventional euclidean distance and Cosine similarity and found that the performance of both of them is similar on retrieval of similar semantic sounds. This shows that the embeddings obtained from the Siamese Neural Network capture the similarity between clips belonging to the audio events very well and can be used for efficient content-based audio retrieval tasks.

6. REFERENCES

[1] Anurag Kumar, Rita Singh, and Bhiksha Raj, "Detecting sound objects in audio recordings," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*. IEEE, 2014, pp. 905–909.

[2] Liwei Wang, Yin Li, and Svetlana Lazebnik, "Learning deep structure-preserving image-text embeddings," pp. 5005–5013, 06 2016.

[3] Eng-Jon Ong, Syed Husain, and Miroslaw Bober, "Siamese network of deep fisher-vector descriptors for image retrieval," 02 2017.

[4] Yonggang Qi, Yi-Zhe Song, Honggang Zhang, and Jun Liu, "Sketch-based image retrieval via siamese convolutional neural network," pp. 2460–2464, 09 2016.

[5] Erling Wold, Thom Blum, Douglas Keislar, and James Wheaten, "Content-based classification, search, and retrieval of audio," *IEEE multimedia*, vol. 3, no. 3, pp. 27–36, 1996.

[6] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-based music information retrieval: Current directions and future challenges," *Proceedings of the IEEE*, vol. 96, no. 4, pp. 668–696, April 2008.

[7] Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 2, no. 1, pp. 1–19, Feb. 2006.

[8] Jonathan T Foote, "Content-based retrieval of music and audio," in *Multimedia Storage and Archiving Systems II*. International Society for Optics and Photonics, 1997, vol. 3229, pp. 138–148.

[9] Ie Eugene Rehn Martin Chechik, Gal, Samy Bengio, and Dick Lyon, "Large-scale content-based audio retrieval from text queries," in *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2008, MIR '08, pp. 105–112, ACM.

[10] N. M. Patil and M. U. Nemade, "Content-based audio classification and retrieval: A novel approach," in *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)*, Dec 2016, pp. 599–606.

[11] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, pp. 1733–1746, 2015.

[12] D. Ellis, "Robust landmark-based audio fingerprinting," 09 2009.

[13] Jaap Haitsma and Ton Kalker, "A highly robust audio fingerprinting system.," in *Ismir*, 2002, vol. 2002, pp. 107–115.

[14] Avery Wang et al., "An industrial strength audio search algorithm.," in *Ismir*. Washington, DC, 2003, vol. 2003, pp. 7–13.

[15] Barry G Sherlock, DM Monro, and K Millard, "Fingerprint enhancement by directional fourier filtering," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 141, no. 2, pp. 87–94, 1994.

[16] Pedro Cano, Markus Koppenberger, and Nicolas Wack, "Content-based music audio recommendation," in *Proceedings of the 13th annual ACM international conference on Multimedia*. ACM, 2005, pp. 211–212.

[17] Courtenay V Cotton and Daniel PW Ellis, "Audio fingerprinting to identify multiple videos of an event," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2386–2389.

[18] James P Ogle and Daniel PW Ellis, "Fingerprinting to identify repeated sound events in long-duration personal audio recordings," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*. IEEE, 2007, vol. 1, pp. I–233.

[19] Xiaolong Wang and Abhinav Gupta, "Unsupervised learning of visual representations using videos," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2794–2802.

[20] Jonas Mueller and Aditya Thyagarajan, "Siamese recurrent architectures for learning sentence similarity.," in *AAAI*, 2016, pp. 2786–2792.

[21] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah, "Signature verification using a" siamese" time delay neural network," in *Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[22] Colin Raffel and Daniel PW Ellis, "Large-scale content-based matching of midi and audio files.," in *ISMIR*, 2015, pp. 234–240.

[23] Neil Zeghidour, Gabriel Synnaeve, Nicolas Usunier, and Emmanuel Dupoux, "Joint learning of speaker and phonetic similarities with siamese networks.," in *INTERSPEECH*, 2016, pp. 1295–1299.

[24] Ke Chen and Ahmad Salman, "Extracting speaker-specific information with a regularized siamese deep network," in *Advances in Neural Information Processing Systems*, 2011, pp. 298–306.

[25] Raia Hadsell, Sumit Chopra, and Yann LeCun, "Dimensionality reduction by learning an invariant mapping," in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*. IEEE, 2006, vol. 2, pp. 1735–1742.

[26] Karol J Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2015, pp. 1–6.

[27] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *22st ACM International Conference on Multimedia (ACM-MM'14)*, Orlando, FL, USA, Nov. 2014.

[28] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "TUT database for acoustic scene classification and sound event detection," in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, Budapest, Hungary, 2016.

[29] C. Buckley and E. Voorhees, "Retrieval evaluation with incomplete information," in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2004, pp. 25–32.