Metric Learning with Sequence-to-sequence Autoencoder for Content-based Music Identification

Pasindu Wijesena[1], *Lakshman* Jayarathne[1], *Manjusri* Wickramasinghe[1], *Shakya* Abeytunge[1,⋆], and *Pasindu* Marasinghe[1]

[1] *University of Colombo School of Computing, 35, Reid Avenue, Colombo, Sri Lanka*

Abstract. Content-based music identification is an active research field that involves recognizing the identity of a musical performance embedded within an audio query. This process holds significant relevance in practical applications, such as radio broadcast monitoring for detecting copyright infringement. Various approaches for content-based music identification have been explored in the existing literature, yielding diverse levels of performance. However, despite the considerable attention dedicated to this area, no attempts have been made to leverage the dynamical nature of musical works coupled with the modern advances in machine learning such as metric learning for content-based music identification. In this paper, we propose a novel approach that encodes the dynamic nature of musical performances into the latent space of a sequence-to-sequence auto-encoder network. The learning objective is further enforced with the metric learning for music similarity measurement. The proposed model is extensively evaluated by testing it with 14 distortions of the same musical performance. The experimental results demonstrate a substantial increase of 31.71% in hit-rate over the baseline established using related work found in the literature. These findings highlight the potential of our approach to significantly improve content-based music identification, thereby offering promising applications in various practical scenarios.

1 Introduction

In the present context, people utilize applications such as "Shazam"[1] [1] to address the requirement of unveiling the identity of an unfamiliar song or musical performance by retrieving its important metadata such as the title, artist, originated country, etc. Applications like Shazam accomplish this task by establishing a comparison of similarity between the queried audio segment and a curated collection of known audio samples of musical performances stored within their databases. This overarching procedure is commonly denoted as content-based music identification.

The field of content-based music identification is a research area that has become widely investigated. This investigation is inspired not only by its pragmatic applications within society but also by its value in monitoring mechanisms. The automated process of content-based music identification plays a pivotal role in monitoring radio broadcasts, records, CDs, streaming media, and peer-to-peer networks for ensuring adherence to copyright regulations and licensing agreements [2]. To serve these purposes effectively, a trained content-based music identification system should be resilient against deformations such as compression artifacts introduced by various encoding systems, noise introduced during transmission, and subtle alterations in frequency and speed [3].

Utilizing high-dimensional raw audio data for content-based music identification poses challenges and incurs significant computational expenses. This is due to the requirement of computing similarities between every pair of audio samples. One is the query audio, and the other is drawn from a collection of audio samples from recognized musical performances. Consequently, it is more convenient to employ a compact audio clip embedding, commonly referred to as a fingerprint, which contains vital information that differentiates the embedded musical performance within the audio clip from other performances.

Various techniques have been explored for audio fingerprinting by researchers [1, 4–8], including hand-crafted rule-based algorithms, computer vision-based algorithms, and, more recently, data-driven algorithms. Data-driven algorithms incorporate machine learning techniques for fingerprint generation, with Convolutional Neural Networks (CNN) being the most prominent approach [7, 8]. However, only limited research has been conducted considering a musical performance as a dynamic system.

A system that changes over time is called a dynamical system. Since audio waves are closely connected to time, it is possible to observe musical performances as dynamical systems. Learning the underlying principles governing the progression of a musical performance, termed as dynamics, can be utilized to differentiate it from other performances in order to implement content-based music identification techniques. Even though the dynamics of

[⋆] e-mail: jsh@ucsc.cmb.ac.lk

[1] https://www.shazam.com/

musical sequences have been studied since 1990 [9], researchers faced limitations in achieving effective results due to technological constraints. Recently, music dynamics have been employed for measuring melodic similarity between two audio recordings using a generative RNN model [10] and for human speech recognition [11]. However, no work has been found that specifically models the dynamics of a musical sequence for content-based music identification.

The research conducted by B. Suárez *et al.* [11] for human speech recognition introduces a sequence-to-sequence autoencoder model for generating fingerprints from Mel Frequency Cepstral Coefficients (MFCC) features extracted from speech audio signals. The autoencoder model is designed to learn and capture the underlying dynamics of the input sequence to reproduce the corresponding output accurately. However, this approach yielded suboptimal results. Building upon this work, the current research proposes a novel method by integrating metric learning techniques into the baseline approach, aiming to enhance the accuracy and effectiveness of the identification process.

The rest of the paper is organized as follows: In Sec. 2, we delve into related work in content-based music identification. Sec. 3 presents the proposed model and its design. Sec. 4 provides important implementation details and showcases experimental results. Finally, Sec. 5 summarizes and concludes the work.

2 Background and Related Work

2.1 Content-based Music Identification Systems

In early literature, domain knowledge in music and concepts in signal processing were employed to develop audio fingerprinting algorithms. For instance, J. Haitsma *et al.* [4] utilized a comprehensive set of features, including Fourier coefficients, MFCC, and others, with dimensionality reduction techniques. While the model showed resilience against noise additions, re-sampling, and band-pass filtering, it was susceptible to time-scale changes, linear speed alterations, and various encoding schemes. C. Wang *et al.* introduced an industrial-strength audio search algorithm, which served as the foundational algorithm for Shazam [1]. Starting from a regular Short Time Fourier Transform (STFT) spectrogram, the algorithm identifies spectral peaks and generates fingerprints by considering pairs of these peaks. Notably, this algorithm demonstrates robustness against quantization effects, filtering, noise, and significant compression. Later, J. Six *et al.* [5] modified the idea of Wang *et al.* by using Constant-Q spectrogram to achieve invariance to pitch shifting and triplets of spectral peaks rather than pairs to form fingerprints to achieve invariance to time-scale modulation.

Another approach is employing computer vision techniques for content-based music identification by treating the spectrogram generated from a raw audio document as an image and performing object recognition. The music identification system introduced by Y. Ke *et al.* [6], which uses Haar wavelet-like filters to create fingerprints, was only robust for physically introduced noise. X. Zhang *et al.* introduced a robust music identification system [12] that applies the Scale Invariant Feature Transform (SIFT) algorithm on top of an STFT spectrogram that has been quantized and logarithmically scaled to extract scale-invariant feature vectors. This approach has shown better robustness compared to the prior methods.

Machine learning-based techniques for music identification have demonstrated better accuracy. B. Arcas *et al.* introduced a low-powered music identification system [7] integrated into the Google Pixel phone[2] lineup, which employs a CNN to create fingerprints and is trained using metric learning with triplet loss. In contrast, B. Suárez *et al.* utilized an Long Short Term Memory (LSTM)-based sequence-to-sequence autoencoder architecture to generate audio fingerprints from MFCC spectrograms. The model is designed to capture a representation from the query audio documents, enabling it to reproduce the query document frame by frame, and it achieved its best performance when trained with the autoencoder reconstruction error.

Recent researchers have utilized contrastive learning to enhance the effectiveness of audio fingerprinting. For instance, Z. Yu *et al.* introduced a self-supervised methodology for audio fingerprinting based on contrastive learning [8]. The authors conducted evaluations using different processing units and achieved impressive performance with the VGG-16 network variant. However, the performance of this proposed algorithm has not been compared to state-of-the-art music identification algorithms. Additionally, recent studies [2, 13, 14] that also utilize contrastive learning have assessed their approach under a limited set of deformations, without including pitch and temporal changes in their evaluations.

In summary, the approach to music identification has gradually shifted towards the utilization of machine learning with neural networks. However, recent works based on contrastive learning have not been thoroughly assessed for their robustness against a range of distortions. Meanwhile, fewer studies delve into the exploration of learning the dynamics of musical performances within a latent space to enable similarity comparison for music identification [11].

2.2 Metric Learning and Triplet Loss

The content-based music identification problem cannot be modeled as a typical multi-class classification problem due to the regular release of new music. Therefore, in the literature [7, 8], the common approach is to compute the pairwise similarity between query fingerprints and reference fingerprints for identifying the music performance from which the query is extracted. Metric learning is a field of study that addresses this problem. Metric learning methods are utilized to learn distance metrics for measuring the dissimilarity between data points, often utilizing a neural network architecture with two or more identical sub-networks that share parameters [15].

Metric learning with triplet loss is one such method found in the literature [7, 8, 15]. This approach involves

[2] https://pixel.google/business/products

working with triplets of data points, which consist of an anchor data point, a positive data point, and a negative data point. The triplet loss aims to minimize the distance between the anchor and the positive data point while maximizing the distance between the anchor and the negative data point.

3 Proposed Method

In this work, the method proposed by B. Suárez *et al.* for the identification of human-speech audio is applied to the content-based music identification task [11]. The resulting audio fingerprinting model is further improved by applying triplet loss to the learning objective of the baseline model. Finally, a simple matching strategy is proposed to perform music identification using the generated fingerprints.

3.1 Input Features

Constant-Q Transform (CQT) features extracted from digital music audio recordings and converted into binary representations serve as the input for the system. These CQT features are preferred due to their musically inspired algorithm, which enhances musical characteristics while maintaining a low-level representation [16]. A Hanning window is applied to audio documents sampled at 22,050Hz with a hop length of 512 frames for the CQT extraction algorithm to result in 84 frequency bins (12 bins per octave) per frame of the time-series. At each frame, the frequency bin with the highest magnitude is made 1.0 and other frequency bins are made 0.0, resulting in the binary CQT features. This design choice is influenced by the work of Haitsma *et al.* [4], which aims to reduce noise and incorporate cross-entropy loss into the learning objective described in Sec. 3.3. A contiguous subsequence of the binary CQT time series, comprising n frames, is treated as a single input for the audio fingerprinting model.

3.2 Audio Fingerprinting Model Architecture

The sequence-to-sequence autoencoder architecture comprises of two stacked LSTM components in the encoder and decoder, coupled with a single-layer perceptron projection head and a softmax filter in the decoder. The encoder and decoder LSTM components share a common set of parameters. In this architecture, an input sequence is fed into the encoder, and the decoder reconstructs this sequence. The entire model is trained based on the learning objective described in Sec. 3.3. The decoder component is only used during the training phase of the model. After the model is trained, the $(n-1)^{th}$ hidden state $\boldsymbol{H_{n-1}}$, corresponding to an input sequence of n frames (indexed from 0 to $n-1$), is considered the fingerprint for that particular input sequence. Therefore, the dimension (d) of the fingerprints is determined by the dimensionality of the hidden state $\boldsymbol{H_{n-1}}$. Fig. 1 shows the model architecture. The complete network is parameterized by $\boldsymbol{\theta}$, and $\boldsymbol{\theta}$ is learned during the training phase.

3.3 Learning Objective

The audio fingerprinting model is trained by fusing two training objectives. The first objective, termed autoencoder loss (L_{ae}), imposes on the model the ability to capture the underlying dynamics of the input sequence to reconstruct the sequence back from the bottleneck point of the encoder [11]. The second objective, termed triplet loss (L_{tp}), inspired by the work done by F. Schroff *et al.* for face recognition, enforces the model to learn an embedding space that can discriminate between matching recordings and non-matching recordings of musical performances [17].

3.3.1 Autoencoder Loss

Let $x_{i,k}$ be the k^{th} scalar of the d-dimensional binary CQT feature vector $\boldsymbol{x_i}$ ($d = 84$), corresponding to the i^{th} frame of an input sequence $\boldsymbol{x}$. Then, the autoencoder loss for a single data sample is defined by Eq. (1). From the perspective of a single frame, the loss function can be considered the typical cross-entropy loss used in classification tasks. Finally, L_{ae} is defined to be the average of l_{ae} computed for all samples $\boldsymbol{x}$ of a mini-batch.

$$\underset{\theta}{l_{ae}} = -\sum_{i=0}^{n-1}\sum_{k=0}^{d-1} x_{i,k} \cdot \log \hat{x}_{i,k} \tag{1}$$

3.3.2 Triplet Loss

The triplet loss is calculated by considering triplets of data samples. Let $T = (\boldsymbol{x^a}, \boldsymbol{x^+}, \boldsymbol{x^-})$ represent a triplet, with $\boldsymbol{x^a}$ as the anchor, $\boldsymbol{x^+}$ as the positive, and $\boldsymbol{x^-}$ as the negative sample. A positive sample matches the anchor and is generated by distorting the anchor sample. In contrast, a negative sample does not match the anchor. A negative sample is a non-matching sample to the anchor sample.

Let $\boldsymbol{H^a_{n-1}}, \boldsymbol{H^+_{n-1}}, \boldsymbol{H^-_{n-1}}$ be the fingerprints corresponding to the triplet. Let $d(\boldsymbol{A}, \boldsymbol{B})$ be the inverse cosine similarity between two vectors $\boldsymbol{A}$ and $\boldsymbol{B}$, defined by Eq. (2), where ϵ is an arbitrarily small constant, used to stabilize $d(\cdot)$. Then l_{tp} for the triplet T is defined by Eq. (3). γ is an arbitrary constant used to prevent the collapse of triplet loss into a trivial solution. Then, semi-hard triplets are identified from each mini-batch. A semi-hard triplet is a triplet T where $d(\boldsymbol{H^a_{n-1}}, \boldsymbol{H^+_{n-1}}) - d(\boldsymbol{H^a_{n-1}}, \boldsymbol{H^-_{n-1}}) < \gamma$. Finally, L_{tp} is defined as the average of l_{tp} calculated for all identified semi-hard triplets.

$$d(\boldsymbol{A}, \boldsymbol{B}) = \frac{||\boldsymbol{A}||||\boldsymbol{B}||}{\boldsymbol{A} \cdot \boldsymbol{B} + \epsilon} \tag{2}$$

$$\underset{\theta}{l_{tp}} = \max(0, d(\boldsymbol{H^a_{n-1}}, \boldsymbol{H^+_{n-1}}) - d(\boldsymbol{H^a_{n-1}}, \boldsymbol{H^-_{n-1}}) + \gamma) \tag{3}$$

3.3.3 Overall Objective

The general loss function is obtained through a linear combination of L_{ae} and L_{tp} that fuses the two objectives. It is defined in Eq. (4), where α is a hyperparameter of the model ($0 \leq \alpha \leq 1$).

$$\underset{\theta}{L} = \alpha \underset{\theta}{L_{ae}} + (1-\alpha)\underset{\theta}{L_{tp}} \tag{4}$$

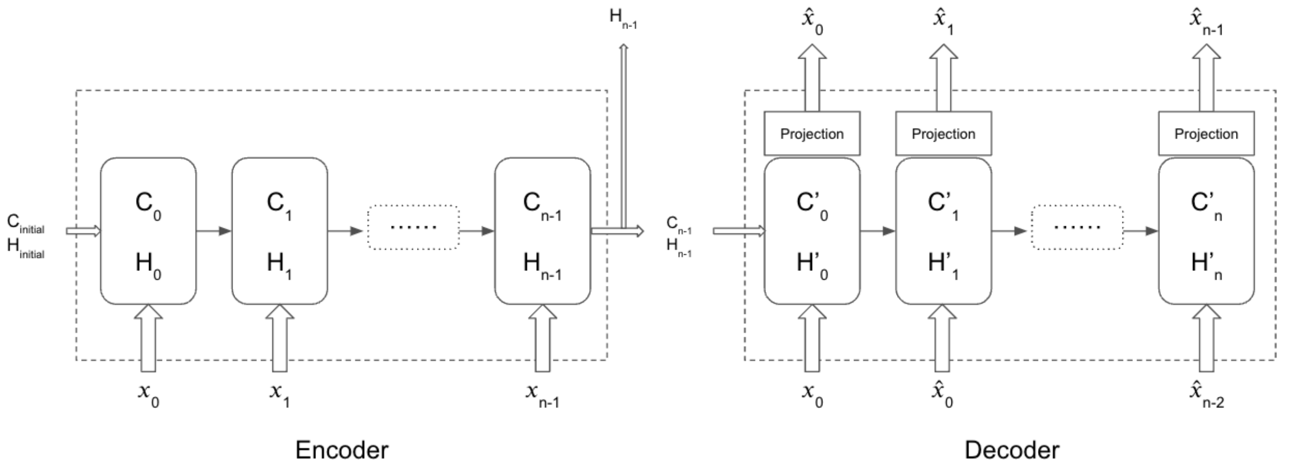

Figure 1. Sequence-to-sequence autoencoder model architecture: $\boldsymbol{C_t}$ is the t^{th} cell state, and $\boldsymbol{H_t}$ is the t^{th} hidden state of the encoder LSTM component. $\boldsymbol{C'_t}$ is the t^{th} cell state, and $\boldsymbol{H'_t}$ is the t^{th} hidden state of the decoder LSTM component. $\boldsymbol{x_i}$ is the i^{th} frame of the input time-series x. $\boldsymbol{\hat{x}_i}$ is the i^{th} frame of the time-series reconstructed by the decoder.

3.4 Music Identification

To perform music identification, the system calculates similarity scores between query audio documents and reference audio documents using fingerprints generated for both. The audio fingerprinting model takes a fixed-length time series (consisting of n frames) as input to produce a single fingerprint. Consequently, a single audio document corresponds to multiple overlapping fingerprints generated from different offsets of the audio signal ($hoplength = 2frames$ is used). For each fingerprint of a query audio document, the system calculates the nearest fingerprint from the reference audio documents and assigns a vote to the corresponding reference audio document. This process is repeated for all the fingerprints generated for a query. Finally, the reference audio document with the highest number of votes is selected as the matching audio document for that specific query.

3.5 Performance Evaluation

The performance of the music identification system is assessed using the hit rate metric as widely used in the literature [8, 14]. The calculation detailed in Eq. (5). To compute this metric, a series of queries is conducted using the music identification system. Subsequently, the system compares these queries with its internal reference database and provides an estimated matching audio document, as described in Sec. 3.4.

$$HitRate = \frac{Number\ of\ correctly\ matched\ queries}{Total\ number\ of\ queries} \tag{5}$$

4 Experimental Settings and Results

4.1 Dataset

In this work, two publicly available music audio datasets are used to train and evaluate the proposed method. The first dataset is the Free Music Archive (FMA) dataset [18], which contains 106,574 audio documents of musical performances of 16,341 artists. Here, a subset containing 1,229 audio tracks from the original dataset is used here for training and validation of the proposed approach. The second dataset is the Covers80[3] dataset is used for evaluation. It contains 160 audio documents of musical performances corresponding to 80 different musical works.

4.2 Data Augmentation

Content-based music identification systems are expected to exhibit robustness to non-musical changes. This robustness is introduced to the proposed music identification system by augmenting the original dataset with distorted audio documents and by learning an embedding space that is invariant to the types of distortion. As a result, original audio documents are distorted by adding random noise and altering the pitch and speed. The levels of distortion applied in each type are presented in Tab. 1.

Table 1. Audio distortion methods applied on original audio documents

Distortion	Magnitude(s)
Random Noise	5%, 10%, 20%, 40%, 50%
Pitch (semitones)	-3, -2, -1, +1, +2, +3
Speed	-10%, -5%, +5%

4.3 Audio Fingerprinting Model Training

The audio fingerprinting model is trained using the Adagrad gradient descent algorithm [19] with mini-batches. A single mini-batch contains b time-series data samples corresponding to original audio documents, accompanied by

[3] http://labrosa.ee.columbia.edu/projects/coversongs/covers80/

$a \cdot b$ time-series data samples corresponding to the distorted audio documents. Together, a single mini-batch contains $(a + 1) \cdot b$ data samples belonging to $(a + 1)$ combinations of distortion variations. During the training phase, only 5 variations of the 14 variations were used due to limitations in RAM capacity. Therefore $a = 2^5$ for the training phase.

The complexity of the audio fingerprinting model introduces a set of hyperparameters in which the values need to be determined. Tab. 2 shows the hyperparameters and the policy used to select the values for each hyperparameter.

Table 2. Hyperparameters of the audio fingerprinting model

Parameter	**Candidate Values**	**Value Selection Policy**
Learning Rate	0.0001 - 0.1	Bayesian optimization
Autoencoder/Triplet Loss Ratio (α)	0.0 - 1.0	Bayesian optimization
Frames per sample (n)	64, 128, 256, 512	Ablation Study
Embedding dimension (d)	32, 64, 128	Ablation Study
Batch Size (b)	64 - 1024	Maximum fit on RAM
Variants (a)	1 - 2^{14}	Maximum fit on RAM

4.4 Ablation Study

The importance of various decisions made is assessed by comparing the performance of the proposed model against potential alternative decisions. Tab. 3 presents a detailed breakdown of the results. The first aspect examined is the impact of fusing two loss functions using the parameter α. When $\alpha = 0.0$, only the triplet loss is used to train the network. When $\alpha = 1.0$, only the autoencoder loss is utilized to train the network, representing the baseline model [11]. An intermediate value, $\alpha = 0.7189$, is determined using a Bayesian optimization algorithm, proving that the autoencoder loss with the triplet loss leads to improved performance compared to using only one loss function. The proposed fusion method achieves a 31.71% performance improvement over the baseline. Secondly, the best performance is achieved when the number of frames per fingerprint (n) is set to 128. It is also observed that performance continues to decrease when n is set to a value above or below 128. Finally, the experiments show that increasing the embedding dimension (d) and the number of fingerprints per query audio document has a positive impact on the performance of the model.

4.5 Robustness to Distortions

Tab. 4 presents the quantification of robustness of the proposed approach ($\alpha = 0.7189$) in contrast to the baseline ($\alpha = 1.0$) inspired by the work of B. Suárez *et al.* . It is observed that the method proposed in this work outperforms the baseline in every type of distortion evaluated in the experiments. Furthermore, only a subset of types of distortions were seen by the audio fingerprinting model during the training phase, marked with stars on Tab. 4. Therefore, the performance seen with other types of distortions, quantify the generalization of the learned model to other distortions.

Table 3. Results of ablation study

Parameter	**Tested Values**	**Hit-Rate**
Autoencoder Loss / Triplet Loss Ratio (α)	$\alpha = 0.0$	83.28%
	$\alpha = 0.7189$	**87.72%**
	$\alpha = 1.0$ (baseline)	56.01%
Number of frames per sample (n)	$n = 64$	81.32%
	$n = 128$	**87.72%**
	$n = 256$	60.28%
	$n = 512$	25.70%
Embedding Dimension (d)	32 dimensions	85.84%
	64 dimensions	85.98%
	128 dimensions	**87.72%**
Number of fingerprints per query	100 fingerprints	80.40%
	150 fingerprints	82.53%
	200 fingerprints	**87.72%**

Table 4. Accuracy for 14 types of distortions

Distortion	**Proposed** ($\alpha = 0.7189$)	**Baseline** ($\alpha = 1.0$)
speed 1.05*	**99.39%**	75.61%
speed 0.95*	**100.00%**	75.61%
speed 0.90	**100.00%**	74.39%
pitch -3	**46.95%**	9.76%
pitch -2	**79.88%**	11.59%
pitch -1*	**96.95%**	40.85%
pitch +1*	**95.73%**	34.76%
pitch +2	**72.56%**	12.20%
pitch +3	**39.63%**	10.37%
noise 0.5	**97.56%**	70.73%
noise 0.4	**99.39%**	83.54%
noise 0.2*	**100.00%**	93.29%
noise 0.1	**100.00%**	95.12%
noise 0.5	**100.00%**	96.34%
Average	**87.72%**	56.01%

5 Conclusions and Future Work

Very few of the prior machine learning models trained for content-based music identification capture the dynamics of audio documents to measure the similarity between them. These existing attempts have produced suboptimal results and only rely on the reconstruction error of the model. In this work, a self-supervised learning method is proposed that combines the autoencoder architecture with triplet loss for music audio fingerprinting, fusing two learning objectives. The sequence-to-sequence autoencoder architecture encourages the model to learn the dynamics of input sequences by minimizing the reconstruction error of the model, while the application of the triplet loss organizes the embedding space to be invariant to a selected set of distortions. Experiments have demonstrated the superiority of fusing the learning objectives over the independent application of the two learning objectives. Future work includes training and evaluating on a larger dataset with more original audio documents and various types of distortions, exploring more sophisticated fingerprint matching algorithms, and evaluating the performance of existing works on publicly available datasets, such as the Covers80 dataset, to enable fair comparisons between different approaches.

References

1. A. Wang. An Industrial Strength Audio Search Algorithm. In: *Proceedings of the 4th International Society for Music Information Retrieval Conference, ISMIR* (2003).
2. A. Singh, K. Demuynck, and V. Arora. Attention-based audio embeddings for query-by-example. In: *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR* (2022), 52–58.
3. M. Müller. Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications. Springer International Publishing, 2015.
4. J. Haitsma and T. Kalker. A Highly Robust Audio Fingerprinting System. In: *Proceedings of the 3rd International Society for Music Information Retrieval Conference, ISMIR* (2002), p. 9.
5. J. Six and M. Leman. Panako - A Scalable Acoustic Fingerprinting System Handling Time-Scale and Pitch Modification. In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR* (2014).
6. Y. Ke, D. Hoiem, and R. Sukthankar. Computer Vision for Music Identification. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR* **1** (2005), 597–604.
7. B. A. y Arcas et al. Now Playing: Continuous Low-Power Music Recognition. In: *arXiv:1711.10958 [cs, eess]* (2017).
8. Z. Yu et al. Contrastive unsupervised learning for audio fingerprinting. In: *arXiv preprint arXiv:2010.13540* (2020).
9. J. P. Boon, A. Noullez, and C. Mommen. Complex Dynamics and Musical Structure. In: *Interface* **19**.1 (1990), 3–14.
10. T. Cheng, S. Fukayama, and M. Goto. Comparing RNN Parameters for Melodic Similarity. In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR* (2018), 763–770.
11. A. Báez-Suárez et al. SAMAF: Sequence-to-Sequence Autoencoder Model for Audio Fingerprinting. In: *Proceedings of the ACM Transactions on Multimedia Computing, Communications, and Applications, TOMM* **16**.2 (2020).
12. X. Zhang et al. SIFT-Based Local Spectrogram Image Descriptor: A Novel Feature for Robust Music Identification. In: *Journal on Audio, Speech, and Music Processing, EURASIP* **2015**.1 (2015), p. 6.
13. S. Chang et al. Neural Audio Fingerprint for High-Specific Audio Retrieval Based on Contrastive Learning. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP* (2021), 3025–3029.
14. X. Wu and H. Wang. Asymmetric Contrastive Learning for Audio Fingerprinting. In: *IEEE Signal Processing Letters* **29** (2022), 1873–1877.
15. M. Kaya and H. Ş. Bilge. Deep Metric Learning: A Survey. In: *Symmetry* **11**.9 (2019).
16. C. Schörkhuber and A. Klapuri. Constant-Q transform toolbox for music processing. In: *Proceedings of the 7th Sound and Music Computing Conference, SMC* (2010), 3–64.
17. F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2015), 815–823.
18. K. Benzi et al. FMA: A Dataset For Music Analysis. In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR* (2017).
19. J. Duchi, E. Hazan, and Y. Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. In: *Journal of Machine Learning Research* **12**.61 (2011), 2121–2159.