Audio Feature Extraction: Research on Retrieval and Matching of Hummed Melodies

Liu Yang
Dezhou University, Dezhou, Shandong 253023, China
Email: yangl1979@outlook.com

Keywords: humming, music, retrieval and matching, dynamic time warping

Received: April 12, 2024

Humming clips facilitate more intuitive and user-friendly music retrieval. This paper combined the K-means clustering algorithm, back-propagation neural network (BPNN) model, and dynamic time warping (DTW) algorithm for humming music retrieval and matching. Initially, the K-means algorithm was used to narrow down the search scope. Then, the BPNN model was employed to extract the abstract features of the music melody, and the DTW algorithm was used to match these abstract features. In the simulation experiment, the classification ability of the K-means algorithm was verified, and then it was compared with the DTW and BPNN+DTW retrieval algorithms. The results showed that the K-means algorithm had good music segment classification performance. The retrieval algorithms used could retrieve the target music more accurately and stably.

Povzetek: Študija se osredotoča na ekstrakcijo avdio značilnosti za lažje iskanje in ujemanje melodij. Avtor uporablja kombinacijo K-means algoritma, povratno-propagacijsko nevronsko mrežo (BPNN) in dinamično prilagajanje časa (DTW) za izboljšanje točnosti iskanja in ujemanja melodij.

1 Introduction

Music, as a form of artistic expression, transcends cultural and linguistic boundaries, resonating deeply within people's hearts [1]. With the continuous progress in the field of music, there has been a significant increase in the repertoire available on music platforms, providing users with more choices. However, this abundance also increases the difficulty of selection. The primary function of a music platform is its search feature. If a user clearly knows the name of the music he is looking for [2], he can quickly locate the desired music by entering its name. If the user knows music-related keywords, he can also efficiently delineate the scope and reduce retrieval difficulties. The above traditional retrieval methods require users to clearly possess explicit information such as the title, lyrics, and author of the music [3]. Once users do not know this information, the retrieval difficulty will significantly increase. The humming melody-based retrieval method is more intuitive. This method only require the user to have an impression of the music melody and be able to hum a portion of it for successful retrieval, and this retrieval method does not require the user to know the key information of the music, nor does it require the platform to annotate the music with much information [4]. Xing et al. [5] proposed an emotion-driven cross-media retrieval system for Chinese folk images and music. The retrieval effectiveness of the method was demonstrated by the experimental findings. Deng et al. [6] proposed a technique for retrieving music that relies on the dynamics of emotions in music over time and verified the advantages of the multi-dynamic texture model in predicting time-varying music emotions. Cheng et al. [7] put forward a user-information-aware music interest topic model. The experimental studies showed that this model significantly enhanced the search accuracy of existing text-based music retrieval approaches. The performance of the algorithms used in the above literature is presented in Table 1. This paper combined the K-means clustering algorithm, back-propagation neural network (BPNN) model, and dynamic time warping (DTW) algorithm for humming music retrieval and matching. The K-means algorithm was initially used to reduce the search scope, the BPNN model was employed to extract the abstract features of the music melody, and the DTW algorithm was used to match the abstract features. Finally, simulation experiments were conducted.

Table 1: The performance of algorithms in relevant literature.

Literature	Algorithm	Accuracy
Xing et al. [5]	DE-SVM	89.75%
Deng et al. [6]	MDT	83.33%
Cheng et al. [7]	UIA-MIT	85.64%

2 Humming-based music retrieval algorithm

Using humming melodies to search for music is more in line with people's daily intuitive habits and reduces the manual workload of music tagging. The process of retrieving music through humming involves three steps: audio preprocessing, audio melody feature extraction, and melody feature matching [8]. In the extraction of audio

melody features, pitch and duration are usually used as features. In matching melody features, since the user is not a professional musician, the hummed melody is improvisatory and uncertain. Simply speaking, for the same melody, the hummed melody of the user will be different from the standard melody of the database, which leads to some difficulties in feature matching [9]. This paper chooses the clustering and BPNN algorithms to retrieve and match the hummed melody. The clustering algorithm can divide the music melody into categories to narrow the matching range. The BPNN algorithm mines the hidden rules in the melody to improve the matching accuracy.

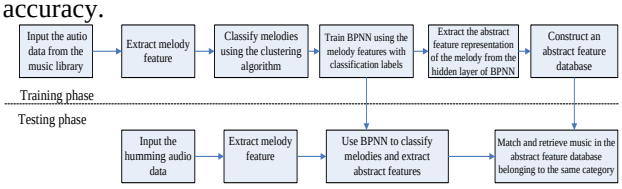


Figure 1: Humming-based retrieval and matching process using BPNN.

The humming-based retrieval matching process using BPNN is shown in Figure 1, and the whole process is divided into the training and testing phases.

The first is the training phase.

① The audio data in the music library undergoes input and preprocessing, which involves procedures such as noise reduction and framing processing [10].

② The melody feature is extracted from the audio. In this paper, the fundamental frequency of the audio frame is selected as pitch, and its calculation formula is:

$$\begin{cases} S(f_i) = \sum_{n=1}^{M} h_n A(nf_i) \\ f_0 = \arg\max\{S(f_i)\} \end{cases}, (1)$$

where f_i is the i-th candidate fundamental frequency, $f_{\max}$ is the global peak frequency in the spectrogram of a frame [11], $S(f_i)$ is the confidence when f_i is utilized as the fundamental frequency, M is the quantity of harmonics, h_n is the compression factor of the n-th harmonic, $A(nf_i)$ is the amplitude of the n-th harmonic when f_i is utilized as the fundamental frequency, f_0 is the calculated fundamental frequency of the signal of a frame, i.e., the frequency with the maximum confidence among the candidate ones. The calculated fundamental frequency of the audio signal of each frame is arranged in time order, which is the pitch sequence [12].

③ The clustering algorithm is used to classify the audio data. This paper uses the K-means clustering algorithm to realize the classification. The steps are as follows. K audio data were randomly selected as cluster centers. Then, the Euclidean distance between the remaining audio data and each cluster center is calculated (the pitch sequence of the audio data is the melody feature vector of the audio, and the Euclidean distance is the distance between the two vectors). The remaining audio data is allocated to different cluster centers according to the nearest principle. Then, the mean vector of all audio melody feature vectors in each cluster is calculated respectively, and the mean vector is used as the new cluster center of the cluster, and the nearest assignment principle is repeated until the cluster center converges to a stable state [13].

④ The category of the cluster to which the audio belongs is used as the melody label of the audio, and then the audio data with melody labels is used to train the BPNN model. The input data of the BPNN model is the melody feature vector of audio. The melody feature vector is forward calculated in the hidden layer of the model, and then the melody label is output in the output layer. The forward calculation formula in the hidden layer is:

$$h = f\left(\sum_{i=1}^{n} \omega x_i - b\right), (2)$$

where h is the hidden state output by the hidden layer, b is the bias term, $f(\cdot)$ is the activation function, and ω is the weight. The melody label output by the output layer is compared with the actual label, and the cross-entropy is used as the error. If there is an error, the parameters are adjusted reversally until the error converges [14].

⑤ The hidden state output of the last hidden layer in the trained BPNN model is used as the abstract feature representation of the melody, and a melody abstract feature database of audio data is constructed.

After that, it is the testing phase.

① The data of humming audio is input and preprocessed by noise reduction and framing.

② The pitch sequence of the data is extracted according to equation (1), which is the melody feature vector.

③ The melody feature vector is input into the trained BPNN model, and the model gives the melody classification label and the abstract feature representation of the melody.

④ According to the melody classification labels given by the BPNN model, the melody of the same class is selected from the melody abstract feature database as the preliminary candidate set. Then, the abstract feature representation of the melody is matched with the melody abstract features in the preliminary candidate set. The DTW algorithm is used to compute the distance between the humming audio and the audio in the preliminary candidate set [15]. The formula is:

$$D = \min_{w(i)} \sum_{i=1}^{M} d[T(i), R(w(i))], (3)$$

where $w(i)$ is the frame number matching path function between the humming audio and the audio in the preliminary candidate set, i is the frame ordinal number of the candidate audio, with a maximum of M frames, $T(i)$ is the melody abstract feature of the i-th frame of the candidate audio, $R(w(i))$ is the melody abstract feature of the humming audio frame, $d[T(i), R(w(i))]$ represents the distance between them, and D is the

distance between the humming audio and the candidate audio. The candidate audio is sorted based on the size of D, from smallest to largest, and the music corresponding to the top k candidate audios is used as the retrieval and matching result according to the requirements.

3 Simulation experiment

3.1 Experimental data

The music datasets used in this simulation experiment were the MIR and MedleyDB datasets. The MIR dataset is a widely used music dataset containing various music types, such as pop, classical, jazz, etc., and the dataset provides audio files and corresponding metadata. The MedleyDB dataset is an audio dataset for music analysis and information retrieval. It contains various styles and types of music, including pop, rock, jazz, classical, etc., and the dataset also provides high-quality audio files and metadata. Non-repeated music was selected from the above dataset, including pop, classical, jazz, rock, and folk, with 100 pieces per type. Each song was divided into several segments. Finally, a total of 12,140 music segments were obtained, 70% of which were used as the training set, and the remaining 30% were used as the test set. The music data had been filtered to remove noise and divided into frames.

3.2 Experimental setup

In the proposed music retrieval and matching algorithm, the K-means algorithm was used to cluster similar music clips, and then the music data with clustering performance was used to train the BPNN model so that it could identify the type of music clips and initially narrow the matching range. The hidden layer of the BPNN model gave the abstract feature representation of the music melody, and the abstract feature representation was used to match music in the music library of the same type obtained after narrowing the matching range. The relevant parameters of the algorithm obtained after the orthogonal experiment are shown in Table 2. The K value of the K-means algorithm part was set to 5. The number of input layer nodes in the BPNN model depended on the number of features in the input sample, i.e., the dimensionality of the melody feature vector. The number of output layer nodes in the BPNN model depended on the initial classification quantity of the K-means algorithm. The reason for using sigmoid in the hidden layer is that it can fit non-linear patterns in data better.

Table 2: The relevant parameters of the proposed algorithm.

Parameter	Setting	Parameter	Setting
The K value of the K-means algorithm	5	The number of iterations in the K-means algorithm	200
The activation function of the BPNN algorithm	sigmoid	The hidden layer of the BPNN algorithm	One layer, 256 nodes
The output layer of the BPNN algorithm	Five nodes	The number of trainings in the BPNN algorithm	500

It was also compared with two other retrieval and matching algorithms. One of them was the DTW algorithm, which directly used the melody feature of extracted pitch sequence to retrieve and match the humming audio, and the k value representing the number of the results was also set to 5. The second algorithm was the BPNN+DTW algorithm. It used the BPNN model to extract the abstract feature representation of the music melody and the DTW algorithm to retrieve and match music based on the abstract feature. The difference between this algorithm and the algorithm proposed in this paper was that the clustering algorithm lies in the absence of a clustering algorithm for classifying audio data used to train the BPNN model.

3.3 Evaluation criteria

Before validating the retrieval matching algorithm, a comparison of the classification performance between support vector machine (SVM) and K-means algorithms was conducted. The evaluation criteria included precision, recall rate, and F value. The corresponding equations are:

$$\begin{cases} P = \dfrac{TP}{TP + FN} \\ R = \dfrac{TP}{TP + FP} \\ F = \dfrac{2 \cdot P \cdot R}{P + R} \end{cases}, \quad (4)$$

where P is the precision, R is the recall rate, F is the comprehensive value of precision and recall rate, TP indicates the true positive rate, FP indicates the false positive rate, FN indicates the false negative rate, and TN indicates the true negative rate.

Precision, recall rate, and F value are commonly used to evaluate retrieval and matching algorithms. However, in addition to the above evaluation indicators, there are two other indicators for retrieval and matching algorithms: the mean reciprocal rank (MRR) and hit rate (HR). They are calculated as follows:

$$\begin{cases} MRR = \dfrac{\sum_{i=1}^{K} (rank_i)^{-1}}{K} \\ HR = \dfrac{\sum_{i=1}^{K} hit_i}{K} \end{cases}, \quad (5)$$

where K is the times of humming, i.e., the times of detections when testing, $rank_i$ is the order of the retrieval

target in the returned list in the i-th retrieval, hit_i is an indication value of whether the search target is in the returned list in the i-th retrieval (1 if it is and 0 if not).

3.4 Experimental results

Firstly, the classification performance of the K-means clustering algorithm was tested and compared with the SVM algorithm (Figure 2). It can be intuitively seen that the K-means clustering algorithm had better classification performance for audio. The precision of the SVM algorithm was 78.9%, the recall rate was 79.6%, and the F value was 79.25%. For the K-means algorithm, the precision was 98.7%, the recall rate was 97.6%, and the F value was 98.15%. The SVM algorithm classified data using the support vector plane, and the data class was determined by which side of the support vector plane the data was on. The distance between the data and the support vector plane was not important, nor was the distance between the data. The K-means algorithm was based on the distance between the data, i.e., the degree of similarity, so the data in the same class were more similar, which was consistent with the intuition.

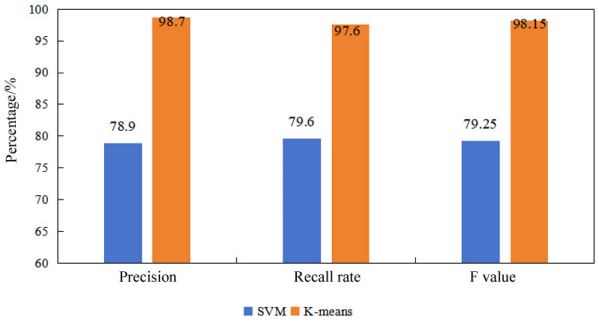
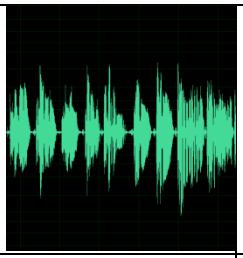


Figure 2: Classification performance of the SVM and K-means algorithm.

Table 3 shows the partial retrieval results of the three retrieval algorithms. It can be seen that all three retrieval algorithms could retrieve the target music, but different retrieval lists were presented. In the retrieval list presented by the retrieval algorithm proposed in this paper, the target music was the first, and the rest of the music was the same in terms of genre. In the search list presented by the BPNN+DTW retrieval algorithm, the target music was in the middle, and the rest had a few different genres. In the search list presented by the DTW algorithm, not only was the target at the bottom of the list, but the genres of other music were also confusing.

Table 3: Partial results of the three retrieval algorithms.

Humming clip		
Music name (genre)	Daybreak (popular)	Five Hundred Miles (ballad)
The search list of the DTW algorithm	1. The Butterfly Lovers (jazz) 2. The Same Smile (jazz) 3. Love is a Song (folk song) 4. The Sailing Ship (rock) 5. Daybreak (pop)	1. Yuan Wu Qu (jazz) 2. Time is a Bulldozer (folk song) 3. In Tune (pop) 4. Sorry for You (ballad) 5. Five Hundred Miles (ballad)
The search list of the BPNN+DTW algorithm	1. A stunning sight (popular) 2. Godfather (popular) 3. Daybreak (popular) 4. Like is Not Love (jazz) 5. Winter Campus (ballad)	1. The Sun (ballad) 2. The Good Wind Will Sing (ballad) 3. Five Hundred Miles (ballad) 4. Come on (popular) 5. The Pilot (jazz)
The search results of the K-means+BPNN+DTW algorithm	1. Daybreak (popular) 2. The Heart Goes Somewhere (pop) 3. In Tune (pop) 4. Manto Manor (popular) 5. Ode to Gallantry (popular)	1. Five Hundred Miles (ballad) 2. Time for a Tear (ballad) 3. Little Flower (ballad) 4. A Leaf Knowing Autumn (ballad) 5. Coming as Promised (folk song)

The performance comparison results of the three retrieval algorithms are presented in Figure 3. It can be seen that the *HR* and *MRR* of the retrieval algorithm using only the DTW algorithm were 0.75 and 0.74; the *HR* and *MRR* of the BPNN+DTW algorithm were 0.84 and 0.86; the *HR* and *MRR* of the K-means+BPNN+DTW algorithm were 0.97 and 0.95. It can be seen that the retrieval and matching algorithm proposed in this paper had the best performance, followed by the BPNN+DTW algorithm, and the DTW algorithm was the worst.

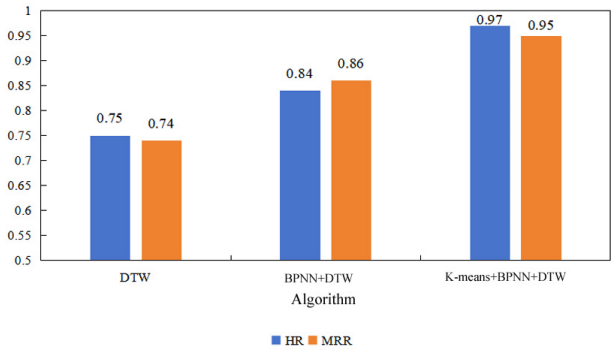


Figure 3: Performance comparison of the three retrieval algorithms.

4 Discussion

The development of music has resulted in an increasing number of music tracks, which makes it more challenging to find the desired music from a vast collection. Traditional methods for retrieving music rely on textual information such as song titles, composers, and lyrics. However, this text-based approach requires users to remember specific details about the music, and not all users are knowledgeable in musical information. In contrast, using humming melodies to search for music is simpler and aligns better with human habits. This article combined the K-means clustering algorithm, BPNN model, and DTW algorithm to retrieve humming melodies. Subsequently, simulation experiments were conducted. Compared to the SVM algorithm, the K-means clustering algorithm performed better in audio classification performance. When the SVM algorithm classified data, it used the support vector. The data category is determined by which side of the support vector plane the data falls on. The distance between data and the support vector plane is not important, nor is the "distance" between data points. On the other hand, the K-means algorithm classified data based on the "distance" or similarity between data points, resulting in more similar data within each category. Afterward, the comparison results of the three retrieval algorithms also showed that the retrieval matching algorithm proposed in this article was the best. The reason for this is that although the DTW algorithm solved the problem of unequal lengths between matched audios by using non-linear folding, it still relied on audio features during the comparison process. The audio feature used solely by the DTW algorithm was pitch sequence, which cannot fully reflect all audio features. By combining the BPNN model with the DTW algorithm, the BPNN model was employed to further abstract feature extraction of melody features, i.e., deepening hidden patterns contained in features, the matching performance was improved. In the combined retrieval and matching algorithm, when training the BPNN model, the K-means clustering algorithm was first used to preliminarily classify audio data and assign labels to them, providing more features for subsequent training of the BPNN model. Therefore, the overall performance of this algorithm was superior. The novelty of this article lies in the use of the BPNN algorithm to extract deep features from audio data, while employing the K-means clustering algorithm to pre-classify the training data, further enhancing the characteristics of audio data and improving the performance of music retrieval and matching.

5 Conclusion

In this paper, the K-means clustering algorithm, BPNN model, and DTW algorithm were combined for humming music retrieval and matching. The K-means algorithm was used to narrow down the retrieval scope, the BPNN model was employed to extract the abstract features of music melody, and the DTW algorithm was used to match the abstract features. In the simulation experiment, the classification ability of the K-means algorithm was verified. Then, the combined algorithm was compared with the DTW and BPNN+DTW algorithms. The results are as follows. (1) The K-means clustering algorithm had good classification performance for audio. (2) Among the three algorithms, the retrieval algorithm proposed in this paper prioritized the target music and retrieved other music of the same genre. (3) The performance of the proposed retrieval and matching algorithm was the best, followed by the BPNN+DTW and DTW algorithms.

References

[1] Schindler A, Rauber A (2016) Harnessing Music-Related Visual Stereotypes for Music Information Retrieval. A*CM Transactions on Intelligent Systems & Technology*, 8, pp. 1-21. https://doi.org/10.1145/2926719

[2] Srinivasa M Y V, Koolagudi S G (2018) Content-Based Music Information Retrieval (CB-MIR) and Its Applications toward the Music Industry: A Review. *ACM Computing Surveys (CSUR)*, 51, pp. 1-46. https://doi.org/10.1145/3177849

[3] Weissenberger L (2015) Toward a universal, meta-theoretical framework for music information classification and retrieval. *Journal of Documentation*, 71, pp. 917-937. https://doi.org/10.1108/JD-08-2013-0106

[4] Furner M, Islam M Z, Li C T (2021) Knowledge Discovery and Visualisation Framework using Machine Learning for Music Information Retrieval from Broadcast Radio Data. *Expert Systems with Applications*, 182, pp. 1-11. https://doi.org/10.1016/j.eswa.2021.115236

[5] Xing B, Zhang K, Sun S, Zhang L, Gao Z, Wang J, Chen S (2015) Emotion-driven Chinese folk music-image retrieval based on DE-SVM. *Neurocomputing*, 148, pp. 619-627. https://doi.org/10.1016/j.neucom.2014.08.007

[6] Deng J J, Leung C H C (2015) Dynamic Time Warping for Music Retrieval Using Time Series Modeling of Musical Emotions. *IEEE Transactions on Affective Computing*, 6, pp. 137-151. https://doi.org/10.1109/TAFFC.2015.2404352

[7] Cheng Z, Shen J, Nie L, Chua T S, Kankanhalli M (2017) Exploring User-Specific Information in Music Retrieval. *Acm Sigir Forum*, 51, pp. 655-664. https://doi.org/10.1145/3077136.3080772

[8] Hu X, Lee J H, Bainbridge D, Choi K, Organisciak P, Downie J S (2017) The MIREX grand challenge: A framework of holistic user-experience evaluation in music information retrieval. *Journal of the Association for Information Science & Technology*, 68, pp. 97-112. https://doi.org/10.1002/asi.23618

[9] Jao P K, Yang Y H (2015) Music Annotation and Retrieval using Unlabeled Exemplars: Correlation and Sparse Codes. *IEEE Signal Processing Letters*, 22, pp. 1771-1775. https://doi.org/10.1109/LSP.2015.2433061

[10] Silva A C M D, Silva D F, Marcacini R M (2022) Multimodal representation learning over heterogeneous networks for tag-based music retrieval. *Expert Systems with Application*, 207, pp. 117969.1-117969.9.

[11] Silva D F, Yeh C, Zhu Y, Batista G E, Keoph E (2019) Fast Similarity Matrix Profile for Music Analysis and Exploration. *IEEE Transactions on Multimedia*, 21, pp. 29-38.

[12] Schwabe M, Heizmann M (2021) Influence of input data representations for time-dependent instrument recognition. *Technisches Messen: Sensoren, Gerate, Systeme*, 88.

[13] Fernández-Rubio G, Carlomagno F, Vuust P, Kringelbach M L, Bonetti L (2022) Associations between abstract working memory abilities and brain activity underlying long-term recognition of auditory sequences. *PNAS Nexus*, 1, pp. pgac216. https://doi.org/10.1093/pnasnexus/pgac216

[14] Lei W H, Lee C Y (2016) Query By Singing/Humming System Using Segment-based Melody Matching for Music Retrieval. *WSEAS Transactions on Systems*, 15, pp. 157-167.

[15] Patil P, Mengade S, Kolpe P, Gawande V, Budhe K (2015) Song Search Engine Based on Querying by Singing/Humming. *International Journal for Scientific Research & Development*, 3, pp. 14-16.