High-Level Libraries for Emotion Recognition in Music: A Review

Yesid Ospitia Medina[1](✉), Sandra Baldassarri[2](✉), and José Ramón Beltrán[2](✉)

[1] ICESI University, Cali, Colombia
yesid.ospitia@gmail.com
[2] University of Zaragoza, Zaragoza, Spain
{sandra, jrbelbla}@unizar.es

Abstract. This article presents a review of high-level libraries that enable to recognize emotions in digital files of music. The main objective of the work is to study and compare different high-level content-analyzer libraries, showing their main functionalities, focused on the extraction of low and high level relevant features to classify musical pieces through an affective classification model. In addition, there has been a review of different works in which those libraries have been used to emotionally classify the musical pieces, through rhythmic and tonal features reconstruction, and the automatic annotation strategies applied, which generally incorporate machine learning techniques. For the comparative evaluation of the different high-level libraries, in addition to the common attributes in the chosen libraries, the most representative attributes in music emotion recognition field (MER) were selected. The comparative evaluation enables to identify the current development in MER regarding high-level libraries and to analyze the musical parameters that are related with emotions.

Keywords: MER (Music Emotion Recognition) · MIR (Music Information Retrieval) · API (Application Programming Interface) · Music features

1 Introduction

Psychology experts have found that music can be considered an emotional transformer [1]. Some studies suggest, in a general way, that musical features as rhythm and harmony have a direct relationship with the listener's emotional perception. The fast rhythms usually generate emotions perceived as positive, meanwhile slow rhythms tend to generate emotions closer to neutral and relaxing ones. In the case of harmony regarding chords progression, the major modes are usually related with a happiness perception, and the minor modes with a sadness perception [2].

That relationship between music and emotions has awakened interest in emotional recognition in music from computational sciences. This task has been addressed from different approaches, such as: labeling processes, application of affective classification models, content analyzers, cultural information gathering and integration and/or physiological signals analysis [3]. Among all these approaches, in this work we will focus on content analyzers due to their direct relationship with the analysis and treatment of sound features.

V. Agredo-Delgado and P. H. Ruiz (Eds.): HCI-COLLAB 2018, CCIS 847, pp. 158–168, 2019.
https://doi.org/10.1007/978-3-030-05270-6_12

With respect to content analyzers, the depth of study that is performed must be specified. On the one hand, there is a study of internal functioning, where one wants to understand, analyze, improve or even propose new techniques, directly related to the processing of the signal or to some later signal processing phases. On the other hand, there are high-level libraries that offer a series of music features extraction functionalities, for which classifying models can be applied, like emotional classification of music.

A content analyzer must rebuild the intrinsic features of music from an initial step of a signal processing block. To make it, various techniques are implemented in the signal, extraction, selection and classification processing algorithm [4]. In Fig. 1 the process that typically follows a content analyzer is shown, from the digital signal processing stage, followed by the extraction of low-level features, and the reconstruction of high-level features, to finally apply classification models to obtain classified musical pieces.

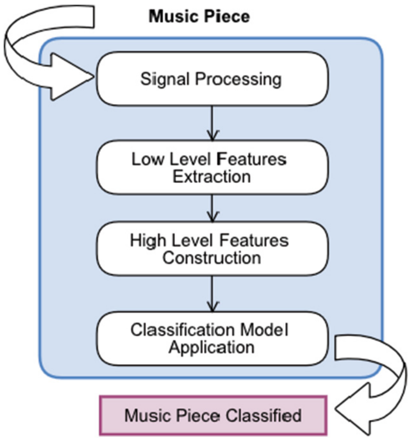

Fig. 1. General process of a content analyzer.

First, content analyzers have the main objective of extracting the sound features. In some cases, some analyzers include additional functionalities, where they have already implemented models that allow classifications by criteria such as musical genre, style, artist, composer, emotions and more [5]. High-level libraries can be considered as those user-friendly content analyzers, which do not require advanced knowledge in topics related to digital signals processing, and which also offer additional classification functionalities.

In this article a selection of high-level libraries, that are part of the content analyzer, is presented. Initially, Sect. 2 presents a basic description of the features of high-level libraries, for later describing, with more detail, the most used ones: *Spotify API* (Sect. 2.1), *jMIR/jAudio* (Sect. 2.2) and *AcousticBrainz* (Sect. 2.3). For each one, the musical features that allow obtaining and reconstructing the emotions that can be recognized are identified, and some works based on those libraries are also presented.

Section 3 establishes a comparative framework between these libraries, evaluating some of the most common and important attributes inside this field of study. Finally, Sect. 4 highlights the conclusions obtained from this study and the proposed future work.

2 High-Level Libraries: Description

The high-level libraries for music features recognition are presented as APIs (*Application Programming Interface*) and are oriented to give a set of functionalities for extraction and classification of intrinsic musical features.

Usually, this type of tools is directed to an end user profile. The reliability and effectiveness of each of these libraries will depend to a large extent on its internal technical definition, which is part of the various techniques developed in the MIR (*Music Information Retrieval*) systems.

Some general features of the different libraries are:

- They have a limit on the variety of intrinsic features of music that can be analyzed.
- They have a certain reliability, in terms of the effective reconstruction of some intrinsic features of the music and their respective classification.
- Regarding their licensing model, they can be free or for commercial use.
- Some are open code, while others simply work like black boxes.
- Some are available as cloud services and require the Internet for their use.
- For the cloud services, there are some access restrictions, for example the number of times a web service can be used per hour.

The use of musical features extraction libraries requires a rigorous selection process to identify the strengths and weaknesses of each of these solutions. It could even be considered the combined use of different libraries, with the purpose of achieving the best possible results.

The sound features that a library can detect are generally classified into two categories: high-level and low-level. In some cases the highest level features of music as rhythm, harmony and mode, are classified inside the tonal and rhythm categories [6]. It is important to understand the type of sound features that each library uses and their direct relationship with musical concepts.

There are different libraries and tools that allow us to work with audio signals for feature extraction: *Spotify API* [7], *jAudio* [8], *AcousticBrainz* [9], *Psysound toolbox from Matlab* [10], *MIR toolbox from Matlab* [11], *Marsyas* [12] and *MediaEval* [13].

In this work, the *Spotify API*, *jMIR* and *AcousticBrainz* libraries have been selected for analysis, taking into account their functional capabilities and the large reference documentation. These libraries can be considered as a starting point for the emotion recognition in music. For these libraries, those high-level features that allow direct identification of emotions in music have been selected. Low-level features have also been selected, which according to [1] can be used to infer emotions.

2.1 Spotify API

Spotify [7] is one of the main platforms for music reproduction. It has a large song repository in different genres and styles, which can be accessed online from a web environment, as from mobile applications. The *Spotify*'s recommender system and the way to classify music to facilitate access and searches are the most relevant features that have allowed the successful reception of users.

Among the various functionalities offered by *Spotify*, there is the possibility of using the services API [7]. This services API started with the *Echonest* project [14], that was absorbed by *Spotify* in 2014, changing some technical aspects of implementation and functionality, among the most significant, the functionality to load sound files disappeared. Nowadays, the *Spotify*'s API only allows to apply its services in songs that are in its repository, being this a big limitation for experimentation exercises.

Next, some of the more relevant functionalities are described:

- To facilitate the integration of web applications and mobile environments to *Spotify*'s services
- To access *Spotify*'s music catalog.
- To obtain information (*metadata*) in *JSON* format about artists, albums, and songs stored in *Spotify*'s catalog.
- Direct access to the playlist created by a user profile.

According to the reference manual presented in the *EndPointReference API* of *Spotify* [7], this library has defined an entity called *audio object features*, with a total of **18 features**. Among them, the most related to emotions in music, are presented in Table 1.

With respect to the recognition of music emotions, there is the possibility of extracting the *valence* and *energy* features. From the point of view of emotional models, this two features can be analyzed as a *arousal*-*valence* type coordinate [15], in such a way that the emotion can be located on a two dimensional plane, including its appreciation (positive or negative) and its intensity level.

Table 1. Some of the high level features extracted by *Spotify API.*

Feature	Description
Key	Song tonality identification
Mode	Song mode (major or minor) identification
Tempo	Associated with rhythm and song's speed. It allows to generate an estimate of beats per minute
Liveness	Allows to identify if it is a live song. Values closer to 1 indicate a higher chance the song is in live
Instrumentalness	Allows to identify if a song have vocal content. The closer its valor to 1 the higher the chance the song does not have vocal content
Energy	Represents intensity and activity level in the song. Values closer to 1 indicate higher intensity in the song
Valence	This feature is associated with the positive a song can be. Closer values to 1 determine positive emotions, meanwhile 0 indicate negative emotions

Regarding the application works, it stands out in [16] and [17] the use of the library *Echonest*, in its first development stages, for the recognition of emotions in audiovisual content, achieving the sound classification in a dimensional model (*arousal-valence*), with some particular adjustments that allow to recognize the following emotions: excited, happy, relaxed, sad, and angry. This model is used by a recommendation system, to facilitate the consumption of audiovisual content through an adaptive *streaming* strategy.

In [18] the library is used to extract audio features with which music is classified according to musical similarity between artists. This information is used by a recommendation system that also analyzes the context information and plots the location of each musical piece in a dimensional model of emotions (*arousal-valence*). In [19] the library is used to analyze the *beat synchronous* feature and generates an emotional classification of music, along with the processing of other features extracted with *Matlab*; the classification process involves the manual labeling of 1000 songs, and then each one of the songs is classified through a dimensional model (*arousal-valence*).

2.2 JMIR/JAUDIO

jMIR [20] is an open code software implemented in Java, that can be used for sound information recovery. It is integrated by a group of components that can be used together, if the experiment conditions allow it. Regarding the recovery of musical information, *jMIR* is constituted as a *framework*, considering its flexibility and the possibility of extension in its uses, in addition to the different components that integrate it an all its advanced functional capacity.

jAudio is one of the components that integrates *jMIR*, and it allows the extraction of sound features from a digital sound file. *jAudio* was designed to work directly with sound in a general way and, at the beginning, it did not implement any specific tool that would allow analyzing high-level musical features. Among the main advantages of *jAudio*, and its main functionalities, the following can be highlighted:

- It works as a local application, so there is no dependency on any type of communication medium.
- It allows to use extension and parameterization by the user. For example, it is possible to create additional sound features to those already defined by default, in order to expand the possibilities of experimentation.
- It is developed in *Java* and can be integrated with other applications.
- It allows to export the recognized features to an *XML* file with all the metadata associated to sound features. This format is interpreted by the different available modules in *jMIR*.
- It can be used together with *jSymbolic*, which is another module that integrates *jMIR*, and allows to analyze more specific features of the musical theory through symbolic information extracted from *MIDI* files.

To obtain emotional type classifications, musical genre and others, *jMIR* has the *ACE2* module that allows to define taxonomies and features, making possible to load in *ACE2* the results obtained with *jAudio* during the low-level features extraction process.

Each one of these results loaded in *ACE2* are considered as an instance, and subsequently various classification models can be applied, obtaining then high-level features that have been previously defined by the user.

jAudio allows the extraction of **26 main low-level features**, from which *metafeatures* can be defined. A *metafeature* is a feature designed by the user, that is calculated from the main features that *jAudio* can extract. In Table 2, some of the low-level features more related with the context of music and emotions are presented.

Table 2. Some of the low-level features extracted by *jAudio*

Feature	Description
Power spectrum	Allows to measure the song's intensity and activity. The signal magnitude is presented in decibels (Db). It is very related with the *energy* feature defined in other libraries
Beat histogram	From a histogram that represents rhythm regularity, the *tempo* of a song can be determined. Other features that allow to calculate the *beat Histogram* are: *Strongest Beat, Beat Sum, Strength of Strongest Beat* [8]. This feature allows to show the frequency with which a determined speed is presented in different moments of a song
Pitch	The frequency range obtained between song's *low pitch* and *high pitch* is a measure that serves as starting point to estimate the predominant note of the song, and with that the tonality and the mode are derived

jMIR also allows integrating classification models, which must be initially designed, and then proceed to load and train them through the *ACE2* component. For the specific case of *jAudio*, although it does not have an emotional classification model, the emotions can be analyzed from some of the low-level features, as the case of *Power spectrum* and *Beat histogram;* which allow to personalize the emotional classification of a song according to its speed and rhythm. There could also be proposed different *metafeatures* that relate *pitch* values, creating associations between tonal and rhythms features and emotions. The emotional classification model could be categorical or dimensional [15], depending largely of the annotation system that is selected to use. The model would need some mechanism to establish the emotional annotation process, and most likely, that mechanism should be implemented as an additional development with which *ACE2* must be integrated, or with any other extern library that allows classifications with predictive algorithms, as the ones used in machine learning techniques.

With respect to the application works, in [8] the *jMIR* functional capacity is described in great detail, in relation to *jAudio*, its broad scope in sound processing problems is a highlighted in a general way. In [2] *jSymbolic* is used to identify and interpret the timbre, rhythm, dynamic and melody. In this work, *jAudio* is used to recognize the *MFCC, Spectral Centroid, Spectral Flux* and *Zero Crossings* features. This work focuses on identifying the emotions in music from chords progression analysis, so that the tonality and mode of the song can be identified and, with that, the emotion can be inferred and placed inside a dimensional model (*arousal-valence*).

2.3 AcousticBrainz

AcousticBrainz [9] is a library whose main functionality is to facilitate the recovery of musical information about songs. *AcousticBrainz* is based on the functionalities offered by *Essentia toolkit* [21], generally include all the sound processing and analysis capacity. This project has been developed by the collaborative effort between the *Music Technology Group* from the *Universitat Pompeu Fabra* and the project *Music Brainz* [22].

AcousticBrainz have classified the features that can extract from a song in two fundamental categories: low-level and high-level. The low-level features include **31 acoustic, 9 rhythmic and 8 tonal descriptors**. In Table 3 shows of the most representative low-level features in the context of music.

Table 3. Some of the low-level features extracted by *AcousticBrainz*.

Feature	Description
Bpm	Associated to the song's rhythm and speed. It allows to generate an estimated of the beats per minute
chords_key	Song's tonality identification
chords_scale	Song's mode (major or minor) identification

AcousticBrainz works with computationally defined and trained models, so from low-level features these models can build high-level features that usually work as classifiers. In terms of high-level features, *AcousticBrainz* uses classification models that allow music to be classified according to various criteria, such as musical genre and emotions. Table 4 shows some of the most representative high-level features in the emotional context.

AcousticBrainz allows the identification of 4 basic emotions [24]: happy, aggressive, sad and relaxed. It is also possible to identify some additional features as with the acoustic, electronic and party case; which could be related with the emotions inside a classification and/or annotation process.

With regards to the *moods_mirex* features, the emotions included inside of each of the *clusters* [23] are detailed next:

- *Cluster 1*: passionate, enthusiastic, confident, bustling, noisy.
- *Cluster 2*: cheerful, animated, funny, sweet, kind/of good character.
- *Cluster 3*: Emotionally intelligent, touching, melancholic.
- *Cluster 4:* humorous, silly, corny, peculiar, capricious, ingenious, ironic.
- *Cluster 5:* aggressive, fervid, tense/anxious, intense, volatile, visceral.

Some works in which this library is applied are: in [25] for the classification of music by musical genre for a database of 120 songs, in [26] is used for the detection of 4 basic emotions through a categorical affective model that recognize emotions: happy, angry, sad, relaxed; additionally also the precision of the emotion recognition is validated, varying the features selection that are extracted and used in the classification model; and finally in [6] the library is used to detect the *valence* and *arousal* values in musical recordings, showing the importance of combining low-level features with high-level features to achieve better results in emotional classifications of music.

Table 4. Some of the high-level features extracted by *AcousticBrainz*.

Feature	Description
mood_happy	Classify the song by happiness emotion. The closer it is to 1 the higher the probability the song is classified with happy emotion
mood_aggressive	Classify the song by aggressiveness emotion. The closer it is to 1 the higher the probability the song is classified with aggressive emotion
mood_relaxed	Classify the song by relaxation emotion. The closer it is to 1 the higher the probability the song is classified with relaxed emotion
mood_sad	Classify the song by sadness emotion. The closer it is to 1 the higher the probability the song is classified with sad emotion
moods_mirex	Classify the song in one of the 5 predefined clusters for emotional state categories (mood) [23]
mood_acoustic	Classify the song in acoustic or not acoustic. The closer it is to 1 the higher the probability the song correspond an acoustic version
mood_electronic	Classify the song according its type of sound. The closer it is to 1 the higher the probability the song has an electronic type of sound
mood_party	Classify the song by type of activity: party. The closer it is to 1 the higher the probability the song classifies with party activity

3 Comparative Framework of Libraries

Table 5 presents a comparison between the three libraries selected in this article: *Spotify API*, *jAudio* and *AcousticBrainz*.

For this comparison, in addition to describing the general features indicated in Sect. 2, the most representative and common attributes in those libraries have been selected, as well as some additional attributes that are considered relevant in *MER* systems and that are mentioned in [27].

About the data presented in Table 5, it is important to highlight that:

- In a general way, high-level libraries have as main functionality the extraction of audio features, however, the quantity, diversity and level of classification (low-level or high-level) differ from one another.
- Some libraries work exclusively with low-level features, so that, they require prior knowledge about the analysis of digital signals, before design the high-level features to be calculated outside the library.
- Not all high-level libraries implement classification models. In some cases, the classifier system must be designed as extension of the libraries, as for example occurs with *jAudio*.
- Typically the high-level libraries use general emotional classification models (*GMER*) [27]. *GMER* consists in applied one same emotional classification of music for all users, and not a personalized classification for each user.
- The *fuzzy* type classification systems are used by some libraries, in some cases through *clusters*, indicating the level of belonging that a certain song has to each *cluster* [27]. Usually, a vector as [0.1, 0.3, 0.9, 0.1] is given as the classifier output, where each position in the vector is associated with a particular *cluster*, and each *cluster* contains a series of emotions.

Table 5. High-level libraries' comparative table

Attribute	Library		
	Spotify API	*jAudio*	*AcousticBrainz*
Type of recognized features	High-level (18)	Low-level (138)	Low-level (80) High-level (20)
Licensing model	Commercial	Free	Free
Results output format	JSON file	XML file	JSON file
Architecture	Cloud service	Local	Cloud service
Allows its extension and parameterization	✗	✓	✗
Allows to implement models for high-level features within the same library	✗	✓	✗
Easy to use (normal, intermediate, advanced)	Intermediate	Advanced	Intermediate
Classification affective model	Dimensional (1D)	✗	Dimensional (2D)
General emotional classification model (GMER)	✓	✓	✓
Recognized emotions	Energy, Valence	✗	Happy, relaxed, sad, aggressive 5 MIREX cluster
Emotional classification of music with fuzzy technique	✗	✗	✓
Recognize musical genre	✗	✗	✓
Recognize live songs	✓	✗	✗
Recognize instrumental songs	✓	✗	✓
Identify the voice by gender	✗	✗	✓
Identify the speed, Bpm	✓	✗	✓
Identify the tonality (Chord recognition)	✓	✗	✓
Identifies song's mode	✓	✗	✓

4 Conclusions and Future Work

The recovery of musical features from digital files implies a solution to the problem of extracting sound properties through libraries and content analyzers. The success of this extraction process depends largely on the effectivity of the internal techniques and the analysis of diverse low-level features combinations, to properly reconstruct high-level features. The high-level features are usually proposed as classification models, being the emotions classification in the music one of the model most used. This classification is complex, and in many cases it must be validated by experts who, through annotation or labeling, associate emotions with a determined song. Later, this manual classification is compared with the automatic classification from the high-level libraries, with the aim of analyzing the success rate, and then, trying to improve this rate using training and learning techniques.

This work has been focused on the classifications of emotions that can be detected through intrinsic music features, and not from the emotions perceived by the listener. For that, the functional capacity of some of the most used high-level libraries was described, and some works in which they have been applied were commented. In addition, a comparative study was carried out among those libraries that contrast their functionalities, including the intrinsic characteristics that they can recognize, the models used to obtain high level parameters and then, in particular, the relationship with the emotions that they can detect.

Based on the study performed, it could be said that although the *MER* libraries currently offer information of interest, *MER* systems are not yet sufficiently developed to offer a universal solution and provide a highly reliable entry for the recommendation systems.

The comparative study developed between the *Spotify API, jAudio* and *AcousticBrainz* libraries is a first step to analyze the musical parameters that are related with emotions. In the future, it is pretended to formulate a classification model that can be used by a recommender system to suggest the listener musical pieces according to the emotion that is intended to transmit. Additionally, it also is pretended to study some additional attributes to the ones verified in Table 5, and that also are not supported by any of the libraries included in this work. The objective of that revision will be to analyze how the attributes mentioned in [27] can improve the emotional classification of music.

Acknowledgment. This work has been partially financed by the Spain Government through the contract TIN2015-72241-EXP.

References

1. Sloboda, J.A.: La mente musical: La psicología cognitiva de la música., Madrid (2012)
2. Cho, Y.-H., Lim, H., Kim, D.-W., Lee, I.-K.: Music emotion recognition using chord progressions. In: 2016 IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 002588–002593. IEEE, Hungary (2016). https://doi.org/10.1109/SMC.2016.7844628
3. Kim, Y.E., et al.: Music emotion recognition : a state of the art review. In: Information Retrieval, pp. 255–266 (2010)
4. Pouyanfar, S., Sameti, H.: Music emotion recognition using two level classification. Proc. Intell. Syst. 1–6 (2014). https://doi.org/10.1109/iraniancis.2014.6802519
5. Mckay, C.: Automatic Music Classification with jMIR, jmir.sourceforge.net (2010)
6. Grekow, J.: Audio features dedicated to the detection of arousal and valence in music recordings. In: 2017 IEEE International Conference on Innovations in Intelligent Systems and Applications (INISTA), pp. 40–44. IEEE, Gdynia (2017). https://doi.org/10.1109/inista.2017.8001129
7. Spotify: Spotify Developer API. https://developer.spotify.com/

8. McEnnis, D., McKay, C., Fujinaga, I., Depalle, P.: JAUDIO: a feature extraction library. In: Proceedings of the International Conference on Music Information Retrieval, pp. 600–603 (2005)
9. Music Technology Group, U.P.F: AcousticBrainz. https://acousticbrainz.org/
10. Cabrera, D., Ferguson, S., Schubert, E.: PsySound3: software for acoustical and psychoacoustical analysis of sound recordings. In: Display, P. (ed.) Proceedings of the 13th International Conference on Auditory Display, pp. 356–363, Canada (2007)
11. Lartillot, O., Toiviainen, P., Eerola, T.: A matlab toolbox for music information retrieval. In: Preisach, C., Burkhardt, H., Schmidt-Thieme, L., Decker, R. (eds.) Data Analysis, Machine Learning and Applications, pp. 261–268. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78246-9_31
12. Tzanetakis, G., Cook, P.: MARSYAS: a framework for audio analysis. Organised Sound **4**, S1355771800003071 (2000). https://doi.org/10.1017/S1355771800003071
13. Soleymani, M., Aljanaki, A., Yang, Y.-H.: DEAM: MediaEval Database for Emotional Analysis in Music, pp. 3–5 (2016)
14. Tristan, J., Brian, W.: Echonest. http://the.echonest.com/
15. Russell, J.A.: A circumplex model of affect. J. Pers. Soc. Psychol. **39**, 1161–1178 (1980). https://doi.org/10.1037/h0077714
16. Solarte, L., Sánches, M., Chanchí, G.E., Duran, D., Arciniegas, J.L.: Dataset de contenidos musicales de video basado en emociones Dataset of music video content based on emotions (2016)
17. Chanchí, G.E.: Arquitectura basada en contexto para el soporte del servicio de vod de iptv móvil, apoyada en sistemas de recomendaciones y streaming adaptativo (2016)
18. Andjelkovic, I., Parra, D., O'Donovan, J.: Moodplay. In: Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization - UMAP 2016, pp. 275–279. ACM Press, Canada (2016). https://doi.org/10.1145/2930238.2930280
19. Soleymani, M., Caro, M.N., Schmidt, E.M., Sha, C.-Y., Yang, Y.-H.: 1000 songs for emotional analysis of music. In: York, A.N. (ed.) Proceedings of the 2nd ACM International Workshop on Crowdsourcing for Multimedia - CrowdMM 2013, pp. 1–6. ACM Press, Barcelona (2013). https://doi.org/10.1145/2506364.2506365
20. JMIR: JMIR Audio Utilities. http://jmir.sourceforge.net/index_jAudio.html
21. Music Technology Group U.P.F: Essentia. http://essentia.upf.edu/documentation/
22. Kaye, R.: Musicbrainz. https://musicbrainz.org/
23. Hu, X., Downie, J.S.: Exploring mood metadata: relationships with genre, artist and usage metadata. In: Proceedings of 8th International Conference on Music Information Retrieval ISMIR 2007, pp. 67–72 (2007)
24. Laurier, C., Meyers, O., Serra, J., Blech, M., Herrera, P.: Music mood annotator design and integration. In: 2009 Seventh International Workshop on Content-Based Multimedia Indexing, pp. 156–161. IEEE (2009). https://doi.org/10.1109/cbmi.2009.45
25. Martins de Sousa, J., Torres Pereira, E., Ribeiro Veloso, L.: A robust music genre classification approach for global and regional music datasets evaluation. In: 2016 IEEE International Conference on Digital Signal Processing (DSP), pp. 109–113. IEEE, Beijing (2016). https://doi.org/10.1109/icdsp.2016.7868526
26. Grekow, J.: Audio features dedicated to the detection of four basic emotions. In: Saeed, K., Homenda, W. (eds.) Computer Information Systems and Industrial Management CISIM 2015, vol. 9339. Springer, Cham. https://doi.org/10.1007/978-3-319-24369-6_49
27. Yang, Y.-H., Chen, H.H.: Music Emotion Recognition. Taylor & Francis Group, Boca Raton (2011)