Music genre classification based on auditory image, spectral and acoustic features

Practical project for the course of Numerical Analysis for Machine Learning

Gabriele Carminati

Summary

1. Abstract

This study proposed in the paper investigates the efficacy of the auditory image approach in music genre recognition. By training various classifiers, the research identifies the Support Vector Machine (SVM) as the most effective. The experiment compares SVM classifier performance using different feature sets, revealing that combining auditory and spectrum features enhances classification.

The goal of this project is to replicate the original experiments, performing more detailed versions of each step when possible. To do so multiple classifiers and multiple datasets are used in each step concluding with a comparison also with a neural network trained on raw data instead of the features extracted and used for the traditional classifiers.

The results comparison seems to suggest that the auditory image features which in the original paper are presented to improve the classification results doesn't always lead to improvement with the results being dependent on the classifier used and on the considered dataset.

2. Paper presentation

The original research paper aimed to assess the effectiveness of the auditory image approach for music genre recognition. To achieve this, the researchers trained various classifiers, identifying the Support Vector Machine (SVM) as the most effective. They then compared the performance of the SVM classifier using different feature sets to determine the most promising ones. They observed that while the features extracted from the auditory image alone performed worse than the features extracted from the spectrum the combination of both lead to an improvement in the classification.

My approach for this project was to reproduce the experiment done by the researchers to verify the accuracy of the results obtained and to confront them also with a classifier built with a neural network and trained directly on the images (spectrum and auditory) rather than on the features extracted from them.

3. Dataset description

The original paper utilized four distinct datasets for the experiment: GTZAN, HOMBURG, ISMIR2004, and GTZAN-NEW. The latter is a modified version of the GTZAN dataset, proposed by Foleis et al. [1], to address issues in the original dataset, such as mislabeling, repetitions, and distortions. Unfortunately, I was unable to find this dataset online, and I decided to exclude it from the experiment, noting that the results obtained from GTZAN were very similar to those from GTZAN-NEW.

3.1. Gtzan

The GTZAN dataset is a well-known dataset in the field of music genre recognition. It is composed of 10 different music genres: blues, classical, country, disco, hip-hop, jazz, metal, rock, reggae and pop.
This dataset is also perfectly balanced, containing 100 samples for each genre. Each sample is a 30 s duration music clip, stored in WAV format having sampled frequency of 22.05 kHz, 16 bits and mono.

3.2. Homburg

The HOMBURG dataset is also well-known in the field. It contains clips from 9 different music genres: alternativa, blues, electronic, folk-country, folk-soul-rnb, jazz, pop, rap-hiphop and rock.
These clips are unequally distributed over 9 genres: alternative (145), blues (120), electronic (113), folk-country (222), funk-soul-rnb (47), jazz (319), pop (116), rap-hiphop (300), rock (504). Audio samples are encoded in MP3 format with a sampling rate of 44.1 KHZ and a bit rate of 128 mbit/s. Each clip has a duration of 10s.

3.3 Ismir2004

The ISMIR2004 dataset contains 1458 full songs distributed 6 genres as follows: classical (640), electronic (229), jazz-blues (52), metal-punk (90), rockpop (203), world (244). Each song has a full duration and is stored as a 44.1 kHz, 16 bits and stereo MP3 file.

4. Dataset preprocessing

To ensure consistency across all datasets for the experiment's following stages, preprocessing was conducted to standardize the data format. Audio files from HOMBURG and ISMIR2004 were converted from MP3 to WAV, transformed from stereo to mono, and their sample rates were reduced from 44.1 kHz to 22 kHz to match the GTZAN dataset's format. Further processing was applied to the GTZAN and ISMIR2004 datasets, where all audio files were segmented into 10-second music clips.

Following the preprocessing stage, each dataset went through three additional processing steps. These steps aimed to extract the essential features from the audio files that would be used to train the classifier and facilitate the experiments outlined in the paper. I will cover each of these steps in the following sections.

4.1. Acoustic features

One set of features extracted from the audio files for the experiments included Mel-Frequency Cepstral Coefficients (MFCCs) and psycho-acoustic music descriptors. Following common practice in the field, only the first 13 MFCC coefficients were extracted from each audio file. The librosa Python library's librosa.feature.mfcc function was employed for this computation.

The following psycho-acoustic music descriptors were also used in the study:

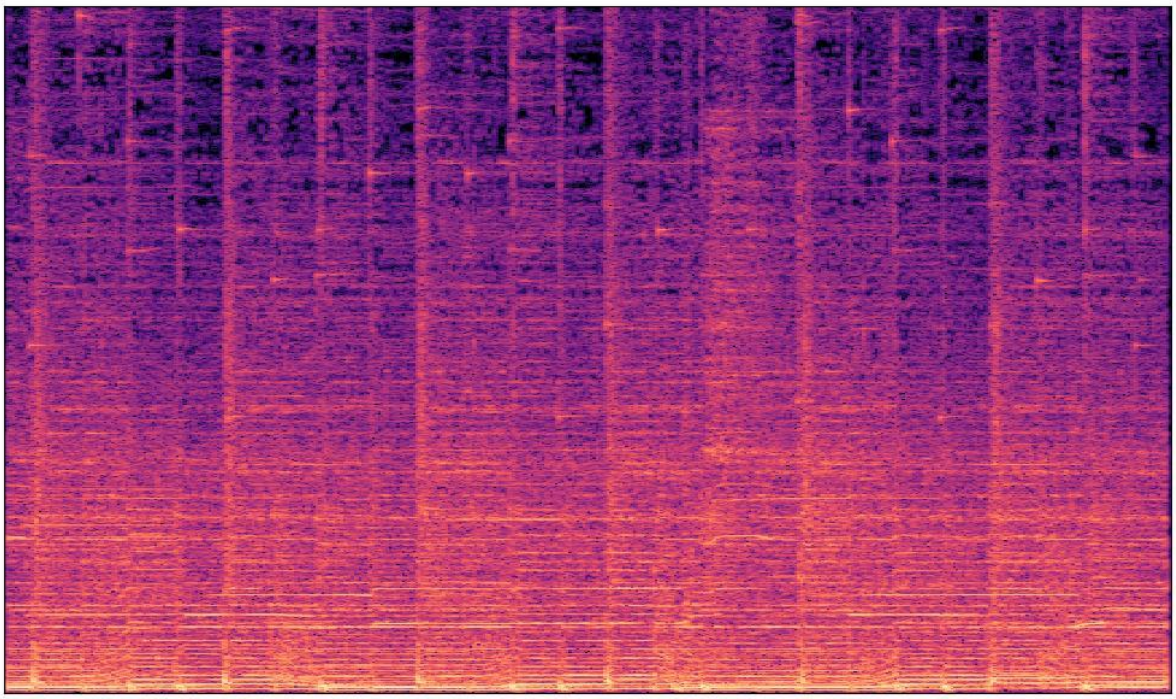
- Statistical Spectrum Descriptor (SSD)
- Rhythm Histogram (RH)
- Modulation Frequency Variance Description (MVD)
- Temporal Statistical Spectrum Description (TSSD)
- Temporal Rhythm Histogram (TRH)

An open-source Python library called rp_extract [2], developed by the Music Information Retrieval Group at TU Wien, was utilized to extract these features from the audio files.

Each extracted feature is a vector of values. For classifier training, these vectors were decomposed into their individual components.

4.2. Spectrum features

Regarding spectral features, the first step involved generating spectrum images from the audio files for all datasets. The GTZAN dataset already included spectrum images, but they were not in logarithmic scale. As the paper and I adopted a logarithmic representation for the spectrum, these images were converted. The librosa and matplotlib libraries were used for spectrum computation.



Spectrum image from a sample of the HOMBURG dataset, genre 'rock'

Once the spectrum images were obtained, the following texture descriptor features were computed:

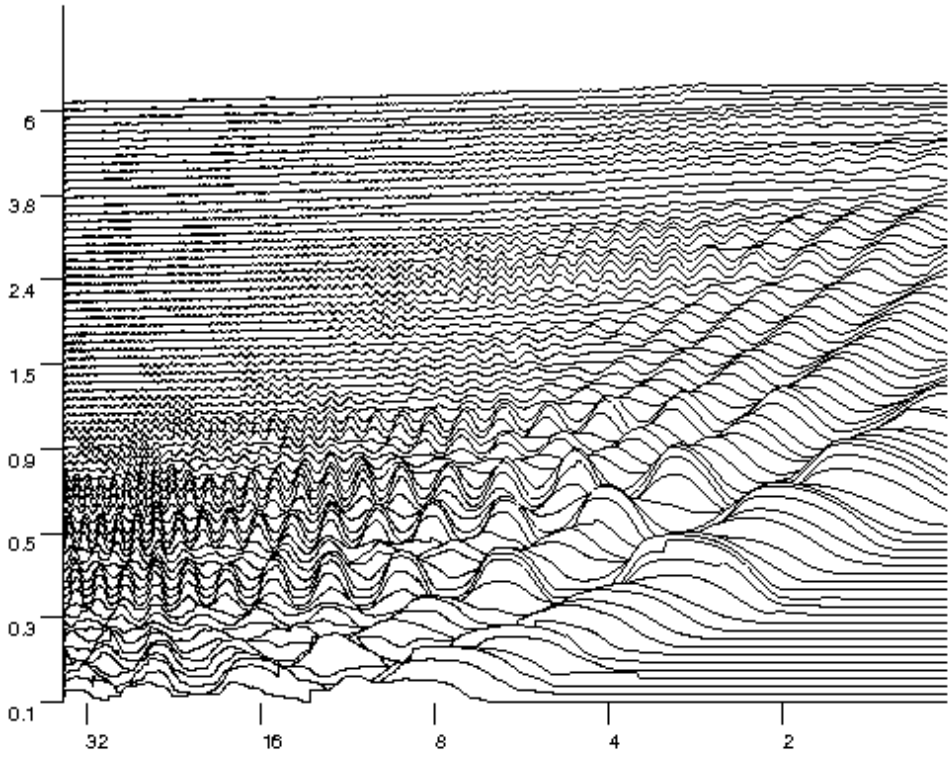
- Local Binary Pattern (LBP): extracted using the local_binary_pattern function from the scikit-image library.
- LBP Histogram Fourier (LBPHF): derived by applying a Fourier transform on the LBP using the NumPy library's built-in fast Fourier transform algorithm.
- Rotation Invariant Co-occurrence Among Adjacent LBPs (RICLBP): computed using a library provided on GitHub by a PhD researcher from the University of Amsterdam [3].
- Local Phase Quantization (LPQ): extracted using publicly available code on GitHub [4].

The original paper also employed another texture descriptor, Dense Sampling Based LBP (DSLBP). Unfortunately, I couldn't locate a Python implementation for this feature and therefore excluded it from the analysis.

4.3 AIM features

The novel feature set introduced in the original paper relies on auditory images, rather than the spectrum images used for the previous features. While the same texture descriptors are computed, the underlying image type differs.

AIM, a time-domain functional model simulating human auditory processing, was employed to generate these auditory images. AIM's processing consists of five key modules: Pre-Cochlear Processing (PCP), Basilar Membrane Motion (BMM), Neural Activity Pattern (NAP), The Identification of The Strobe Points (STROBE), and Stabilized Auditory Image (SAI). The MATLAB library AIM2006 [5] was used to generate auditory images. Once generated, the auditory images were processed identically to the spectrum images for texture descriptor extraction.



Auditory image from a sample of the HOMBURG dataset, genre 'rock'

5. Experiments and evaluation

The researchers in the paper conducted two main experiments, the first one was meant to identify the best classifier between SVM and KNN. The second one wanted to identify the best features to perform the classification, in particular checking if the auditory image is useful or not.

For this work I reproduced the same experiments to confront the results declared in the paper with the one obtained by me.

5.1. Classifier evaluation

To select the optimal classifier, the researchers trained various models on the GTZAN dataset, utilizing one feature at a time from both acoustic and spectrum categories. For a more comprehensive evaluation, I decided to include also the AIM features. In the table below, features labeled with "S-" are derived from the spectrum image, while those labeled with "A-" are from the auditory image. To perform a more complete evaluation I decided to use not only GTZAN, so I trained the classifiers three times, once for each dataset, resulting in the highest accuracy scores for each feature, highlighted in yellow.

- Results for GTZAN dataset

Feature	SVM accuracy	KNN accuracy
MFCC	0.371	0.409
SSD	0.494	0.555
RH	0.444	0.438
MVD	0.324	0.222
TSSD	0.488	0.553
TRH	0.441	0.438
S-LBP	0.349	0.558
S-LBPHF	0.207	0.431
S-RICLBP	0.357	0.558
S-LPQ	0.098	0.555
A-LBP	0.102	0.212
A-LBPHF	0.098	0.200
A-RICLBP	0.102	0.212
A-LPQ	0.227	0.172

- Results for HOMBURG dataset

Feature	SVM accuracy	KNN accuracy
MFCC	0.357	0.365
SSD	0.452	0.402
RH	0.410	0.322
MVD	0.380	0.248
TSSD	0.452	0.402
TRH	0.410	0.322
S-LBP	0.433	0.415
S-LBPHF	0.359	0.375
S-RICLBP	0.441	0.415
S-LPQ	0.269	0.404
A-LBP	0.232	0.253
A-LBPHF	0.232	0.267
A-RICLBP	0.232	0.253
A-LPQ	0.330	0.243

- Results for ISMIR2004 dataset

Feature	SVM accuracy	KNN accuracy
MFCC	0.554	0.639
SSD	0.622	0.771
RH	0.605	0.651
MVD	0.541	0.483
TSSD	0.619	0.771
TRH	0.602	0.651
S-LBP	0.591	0.732
S-LBPHF	0.499	0.603
S-RICLBP	0.592	0.732
S-LPQ	0.365	0.729
A-LBP	0.335	0.454
A-LBPHF	0.335	0.401
A-RICLBP	0.335	0.454
A-LPQ	0.453	0.383

From this analysis, it is evident that the KNN approach outperforms the SVM approach. However, the paper reports contrary results. Consequently, for the following experiments, I used both KNN and SVM. KNN was included as it proved to be the best option based on my

experiments, and SVM was used to maintain consistency with the researchers' results and ensure a valid comparison.

5.2. Features evaluation

After selecting SVM as their classifier, the researchers computed the classification accuracy for each feature on each dataset. Since I have already completed this step, I will not repeat it here. Instead, I will focus on the final experiment conducted in the paper, where the researchers proposed a set of feature combinations and evaluated the accuracy obtained on each dataset. The proposed set of features is as follows:

- A1=SSD+MFCC
- A2=SSD+RH+MFCC
- A3=SSD+MVD+MFCC
- S1=S-LBP+S-LBPHF+S-LPQ
- S2=S-LBP+S-LBPHF+S-RICLBP
- S3=S-LBP+S-LPQ
- S4=S-LBP+S-RICLBP
- S5=S-DSLBP+S-LPQ
- S6=S-LBP+S-RICLBP+S-LPQ
- S7=S-LBP+S-DSLBP+S-LPQ
- F1=A2+S-LPQ
- F2=A2+S3
- F3=A2+S5
- F4=A2+S5+M-LBP
- F5=A2+S5+M-LBPHF
- F6=A2+S5+M-RICLBP
- F7=A2+S5+M-DSLBP
- F8=A2+S5+M-LPQ

As mentioned earlier, I excluded the DSLBP from the descriptors used in the analysis, which necessitated excluding features S5, S7, F3 and F7 as well. To maintain the focus on evaluating features F4 to F8, I replaced the S5 feature with the S3 feature. This substitution was made because S3, derived from the spectrum image, achieved a similar accuracy to S5 in the original paper. In my work I also proposed a new feature combination, which is the result of the combination of the 5 most promising single features:

- G1=SSD+TSSD+S6

In the following table I present the classification accuracy obtained by each feature on each dataset for both the SVM and KNN classifiers:

- Results for GTZAN dataset

Feature	SVM accuracy	KNN accuracy
A1	0.387	0.399
A2	0.389	0.404
A3	0.387	0.396
S1	0.207	0.464
S2	0.376	0.605
S3	0.100	0.575
S4	0.377	0.605
S6	0.371	0.605
F1	0.386	0.403
F2	0.386	0.403
F4	0.386	0.403
F5	0.387	0.406
F6	0.086	0.262
F8	0.386	0.404
G1	0.366	0.603

- Results for HOMBURG dataset

Feature	SVM accuracy	KNN accuracy
A1	0.357	0.378
A2	0.354	0.378
A3	0.357	0.375
S1	0.373	0.399
S2	0.420	0.428
S3	0.283	0.394
S4	0.420	0.428
S6	0.417	0.428
F1	0.357	0.380
F2	0.357	0.380
F4	0.357	0.380
F5	0.357	0.383
F6	0.280	0.328
F8	0.357	0.383

G1	0.412	0.428

- Results for ISMIR2004 dataset

Feature	SVM accuracy	KNN accuracy
A1	0.409	0.500
A2	0.409	0.496
A3	0.409	0.502
S1	0.284	0.415
S2	0.389	0.544
S3	0.175	0.526
S4	0.389	0.544
S6	0.392	0.544
F1	0.409	0.493
F2	0.409	0.493
F4	0.409	0.493
F5	0.409	0.498
F6	0.140	0.279
F8	0.410	0.495
G1	0.392	0.546

These results further confirm that a KNN classifier outperforms an SVM classifier, reinforcing the conclusions from the initial experiment.

Another notable observation is that the combination of features extracted from the auditory image does not seem to enhance the classifier's performance, either individually or when combined with other features. The tables above highlight in yellow the four features that achieved the highest accuracy scores for each classifier. In four out of six instances, these features were S2, S4, S6, and G1.

Interestingly, with an SVM classifier, auditory image features led to a slight improvement in classification accuracy in two out of three cases, while with KNN, this improvement was not observed. This discrepancy might explain the researchers' conclusion that these features were useful. A deeper analysis suggests that their utility may depend on the specific classifier used.

5.3. Neural Network approach

After realizing that the utility of the auditory image varies depending on the classifier used, I decided to test a new approach. I aimed to build a feedforward artificial neural network and feed it directly with auditory and spectrum images to compare its performance with previous classifiers.

The first step involved reducing the image dimensions to prevent overfitting, where the number of features greatly exceeds the number of samples. The resizing factor was dataset-dependent: for ismir2004, I used larger images due to the greater amount of data available compared to gtzan or homburg.

The neural network was a feedforward model with hyperbolic tangent activation functions, and the parameters were initialized using Glorot Normal initialization. A softmax function was applied to the network's last layer to ensure that each neuron's output ranged between zero and one, and that the sum of all outputs was exactly one. This way, the output could be interpreted as the probability of the input belonging to each class. The class with the highest probability is the output of the classification model.

In the following table are reported the accuracy score of the neural network on each dataset and image used.

	GTZAN	HOMBURG	ISMIR2004
Spectrum image	0.610	0.382	0.584
Auditory image	0.328	0.374	0.500

Comparing these results with those from the previous step, we observe that a neural network trained directly on the images achieves slightly better performance than a classifier trained on texture descriptors. The table also highlights the superiority of spectrum images over auditory images in music genre recognition.

6. Conclusions

After replicating the experiments there are some critiques that must be done to the work done by the researchers.

First, the selection of the classifier. From the experiment conducted it seems that the approach that obtained the best results is the use of a KNN classifier instead of an SVM. The choice of this classifier can be the reason that led to the result presented while comparing the results of the two approaches also in the next steps of the experiment conducted by the researchers could have led to different results.

Secondly, in their conclusions the researchers stated that the auditory image approach, when combined with spectrum features, can improve music genre classification. However, after replicating the tests using not only the SVM classifier but also the KNN one and a neural network, it's evident that the utility of auditory image features varies with the classifier used. This approach seems to obtain better results when an SVM is used for the classification while with the other method it's clearly better to leverage only the information obtained by the spectrum and the acoustic features.

Finally, the usage of a neural network trained on the raw spectrum images without any step of texture description seems to slightly outperform the other classification methods proposed. However, the training of the network is also more time demanding with respect to the other classifiers so there are some tradeoffs between accuracy and training time to be considered when choosing the classification method.

To sum up, the results of this work suggest that while auditory images can be useful, their application should be carefully considered in the context of the specific classifier and dataset being used for music genre recognition.

7. References

1. Foleis, J.H., Tavares, T.F.: Texture selection for automatic music genre classification. Appl. Soft Comput. 89, 106–127 (2020). https://doi.org/10.1016/j.asoc.2020.106127
2. Rp extrct library https://github.com/tuwien-musicir/rp_extract
3. RICLBP library https://github.com/mnishant2/RICLBP
4. LPQ library https://gist.github.com/absaravanan/a145f3b1a364d2a499bca79525b2667b
5. AIM2006 library https://www.acousticscale.org/wiki/index.php/AIM2006_Documentation