Music Genre Recognition Using Gabor Filters and LPQ Texture Descriptors

Yandre Costa[1,2], Luiz Oliveira[2],
Alessandro Koerich[2,3], and Fabien Gouyon[4]

[1] State University of Maringá (UEM), Maringá, PR, Brazil
[2] Federal University of Paraná (UFPR), Curitiba, PR, Brazil
[3] Pontifical Catholic University of Paraná (PUCPR), Curitiba, PR, Brazil
[4] Institute for Systems and Computer Engineering of Porto (INESC), Porto, Portugal
yandre@din.uem.br, lesoliveira@inf.ufpr.br,
alekoe@ppgia.pucpr.br, fgouyon@inescporto.pt

Abstract. This paper presents a novel approach for automatic music genre recognition in the visual domain that uses two texture descriptors. For this, the audio signal is converted into spectrograms and then textural features are extracted from this visual representation. Gabor filters and LPQ texture descriptors were used to capture the spectrogram content. In order to evaluate the performance of local feature extraction, some different zoning mechanisms were taken into account. The experiments were performed on the Latin Music Database. At the end, we have shown that the SVM classifier trained with LPQ is able to achieve a recognition rate above 80%. This rate is among the best results ever presented in the literature.

Keywords: Music genre, texture, image processing, pattern recognition.

1 Introduction

In recent years, a huge amount of data from different sources has become available online. In most cases, this information is not organized according to some predefined pattern. Thus, tasks related to automatic search, retrieval, indexing and summarization has become important questions, whose solutions could support a good and efficient access to this content. For some time, textual annotation was used to organize and classify multimedia data. However, this is not a good way to deal with this content efficiently. Textual annotation requires a large amount of human labor and, moreover, is subject to human perception subjectiveness.

Digital music is among the most common types of data distributed through the internet. There are a number of studies concerning to audio content analysis using different features and methods. Automatic music genre recognition is a crucial task for a content based music information retrieval system. As stated by Tzanetakis and Cook in [1], musical genres are categorical labels created by humans to characterize pieces of music. A musical genre is characterized by the

J. Ruiz-Shulcloper and G. Sanniti di Baja (Eds.): CIARP 2013, Part II, LNCS 8259, pp. 67–74, 2013.

common characteristics shared by its members. These characteristics typically are related to the instrumentation, rhythmic structure, and harmonic content of the music. In some studies it was found that genre is an important attribute which helps users in organizing and retrieving music files.

Costa et al. presented in [2] the first results obtained in music genre classification using features extracted from spectrograms. Spectrogram is a visual representation of the spectrum of frequencies in a sound [3]. In the most common representation, spectrogram is a graph with two geometric dimensions: the horizontal axis represents time, the vertical axis is frequency; a third dimension indicating the amplitude of a particular frequency at a particular time is represented by the intensity or color of each point in the image. As shown in Figure 2, texture is the most noticeable visual content in a spectrogram image. Taking this into account, we have explored different texture descriptors presented in the image processing literature in order to capture information to describe this content. In [2], we used the well-known Gray Level Co-occurrence Matrix (GLCM) to capture the textural content from the spectrogram images. By analyzing the spectrogram images, we have noticed that the textures are not uniform, so we decided to consider a local feature extraction beyond the global feature extraction. In that work, only one classifier was created even when a zoning strategy was used in order to preserve local information, and the final decision was done through majority voting among the results obtained with feature vectors extracted from different zones. In [4] and [5], the authors have evaluated the Local Binary Pattern (LBP) texture descriptor trying to capture the spectrogram image content. Furthermore, the authors introduced the creation of one classifier for each created zone, combining their outputs in order to get the final decision using fusion rules presented by Kittler *et al.* [6], like Product, Sum, Max and Min. The best obtained results on the ISMIR 2004 dataset are comparable to the best results described in the literature. Regarding LMD dataset, the best obtained result is the best ever obtained using artist filter.

In this work, we are interested in investigate the performance of LPQ and Gabor filters texture operators in music genre recognition using spectrogram images. The reason for choosing Gabor filters is that in our previous works, there is a lack of experiments using some spectral texture descriptor approach. With regard to LPQ, the choice was done because this is a novel operator which has shown good performance in many different works presented in the literature.

This paper is organized as follows: Section 2 describes the feature extraction performed in this work. Section 3 describes the classification while Section 4 reports the results and discussions about them. Section 5 concludes this work.

2 Feature Extraction

Before proceed the generation of the visual representation, we performed a time decomposition based on the idea presented by Costa et al. [7] in which an audio signal S is decomposed into n different sub-signals. Each sub-signal is simply a projection of S on the interval $[p, q]$ of samples, or $S_{pq} =< s_p, \ldots, s_q >$.

In the generic case, one may extract K (overlapping or non-overlapping) subsignals and obtain a sequence of spectrograms $\overline{Y}_1, \overline{Y}_2, \ldots, \overline{Y}_K$. We have used the same strategy used in [8], which considers three 10-second segments from the beginning ($\overline{Y}_{beg}$), middle ($\overline{Y}_{mid}$), and end ($\overline{Y}_{end}$) parts of the original music. In order to avoid segments that do not provide good discrimination among genres, we decided to ignore the first ten seconds and the last ten seconds of the music pieces. The rationale behind this strategy is that some common effects present in these parts of the music signal, like fade in and fade out, and some kinds of noise, like those produced by the audience, could turn these signal samples less discriminant than the others.

After the signal decomposition, the next step consists in converting the audio signal into a spectrogram. The spectrograms were created using a bit rate = 352kbps, audio sample size = 16 bits, one channel, and audio sample rate = 22.05 kHz. Figure 1 depicts the signal segmentation and spectrogram generation.

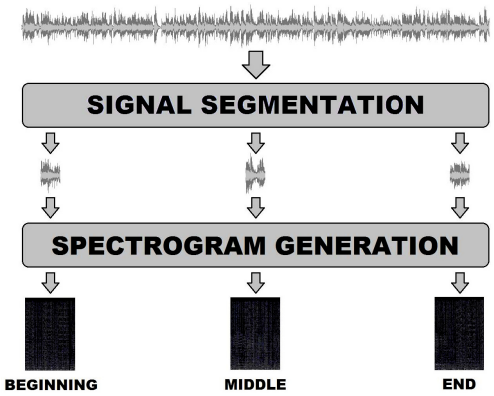

Fig. 1. Creating spectrograms using time decomposition

Once the spectrograms were generated we proceeded the texture feature extraction from these images. As stated before, the approach proposed in this work considers that the main visual content present in the spectrogram images is the texture. With this in mind, we used Gabor filters and LPQ texture operator to capture the image content.

In this work, before proceeding the feature extraction with Gabor filters, the spectrogram images were scaled to 64×64 pixels. Once it was done, the Gabor wavelet transform was applied on the scaled image with 5 different scale levels and 8 different orientations, which results in 40 subimages. For each subimage, 3 moments are calculated: mean, variance and skewness. So, a 120-dimensional vector is used for Gabor texture features. More details about Gabor filters can be found in [9].

Our experiments with LPQ were performed with the original implementation. The window size used to compute the short-term Fourier Transform was

empirically adjusted to 7×7. Additional mathematical details about LPQ can be found in [10].

2.1 Global and Local Feature Extraction

The rationale behind the zoning and combining scheme is that music signals may include similar instruments and similar rhythmic patterns which leads to similar areas in the spectrogram images. By zoning the images we can extract local information and try to highlight the specificities of each music genre.

A positive side effect obtained with zoning strategy is that one can create a specific classifier to deal with the features extracted from each specific zone. Thus, we can naturally obtain several classifiers. Not by chance, the best results achieved in previous works were obtained by combining these classifiers outputs.

In order to proceed the local feature extraction, we have evaluated three different number of linear zones (1,5, and 10), which are applied to the spectrogram image before extracting textural features. Thus, considering that three spectrogram images were generated from each music piece, since we extracted three segments, the number of total zones and consequently the number of classifiers is $3n$, where n is the number of zones per segment. Figure 2 shows a linear zoning scheme, with $n = 10$, superimposed over a spectrogram image extracted from 30 seconds signal (three segments of ten seconds).

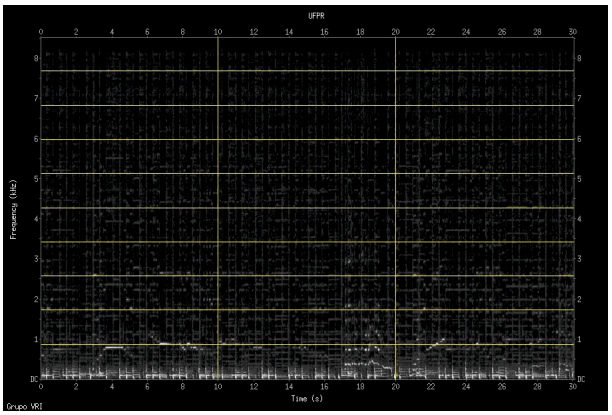

Fig. 2. Linear zoning used to extract local information

3 Classification

The classifier used in this work is Support Vector Machine (SVM), introduced by Vapnik in [11]. Normalization was performed by linearly scaling each attribute to the range [-1,+1]. The Gaussian kernel was used, with parameters C and γ tuned using a greedy search.

The classification process is done as follows: as aforementioned, the three 10-second segments of the music are converted to the spectrograms ($\overline{Y}_{beg}$, $\overline{Y}_{mid}$, and $\overline{Y}_{end}$). Each of them is divided into n zones, according to the values of n described in subsection 2.1. Then, a 120-dimensional Gabor filters feature vector and a 256-dimensional LPQ feature vector were extracted from each zone. Next, each one of these feature vectors is sent to a specific classifier, which assigns a prediction to each one of the ten possible classes. Training and classification were carried out using the 3-fold cross-validation. For each specific zoning scheme, we created $3n$ classifiers with 600 and 300 feature vectors for training and testing, respectively. With this amount of classifiers, we used estimation of probabilities to proceed the combination of outputs in order to get a final decision. In this situation, is very useful to have a classifier producing a posterior probability $P(class|input)$. Here, we are interested in estimation of probabilities because we want to try different fusion strategies like Max, Min, Product, and Sum.

4 Experimental Results and Discussion

Firstly, some details about the music database used in the experiments reported here are described. The Latin Music Database (LMD) is a digital music database created for support research in music information retrieval. The database was presented by Silla et al. [12]. It is composed of 3,227 full-length music samples in MP3 format originated from music pieces of 501 artists. The database is uniformly distributed along 10 music genres.

In our experiments we have used the artist filter [13] restriction when splitting the dataset to create folds. The use of the artist filter does not allow us to employ the whole dataset since the distribution of music pieces per artist is far from uniform. Thus, 900 music pieces from the LMD were selected, which are split into 3 folds of equal size (30 music pieces per class). In order to compare the results obtained here with those obtained in other works, the folds splitting taken was exactly the same used by Lopes et al. [14] and by Costa et al. [2] [4] [5]. The results described here refer to the average recognition rate considering the three folds aforementioned. In addition, the standard deviation between the three folds used in classification is presented.

Table 1 reports the results obtained when features extracted with Gabor filters were used with four different fusion rules and with the three different zoning configurations mentioned in section 2.1. As in the results presented in [4], the best result was obtained when five zones were created. Like in that work, one can see that increasing the number of zones up to a certain point we observe a noticeable performance improvement.

Table 2 presents results obtained using LPQ texture descriptor. Interestingly, the best result with LPQ, both in terms of recognition rates and standard deviation, were obtained when the global feature extraction (without zoning) was used. One can notice that the results obtained with global feature extraction and five linear zones are very close to each other. However, it is important to contrast that using global feature extraction, only three classifiers are created

Table 1. Average recognition rates (%) and standard deviation obtained between the three folds using different number of zones with Gabor filters

Number of zones	Maximum rule	Minimum rule	Product rule	Sum rule
1	55.89±9.94	56.67±11.60	59.78±9.91	58.78±9.08
5	66.22±2.22	69.67±2.33	**74.67±3.79**	74.11±2.69
10	60.56±1.02	65.33±2.85	71.78±1.84	71.00±0.58

whereas 15 are created when five linear zones are created. In addition, the best result obtained with LPQ is very close to, but sligtly better, the best result reported in [4], obtained with Local Binary Pattern (LBP) texture descriptor.

Table 2. Average recognition rates (%) and standard deviation obtained between the three folds using different number of zones with LPQ

Number of zones	Maximum rule	Minimum rule	Product rule	Sum rule
1	76.89±2.12	77.22±1.68	**80.78±0.77**	79.44±1.17
5	74.00±1.91	76.00±1.66	80.67±1.44	80.56±1.10
10	70.11±2.57	73.33±1.25	79.00±0.89	78.00±0.27

4.1 Discussion

Unlike the results obtained with Gabor filters and the texture descriptors used in [4], i.e. LBP and GLCM, the best result with LPQ was obtained using global feature extraction, as shown in table 2. This is very interesting, once with global feature extraction we create a smaller amount of classifiers, which decreases the overall system complexity. In addition, it is important to notice that the result obtained with LPQ is the best one ever obtained with linear zoning or global feature extraction taking into account all the texture descriptors already experimented on the LMD dataset.

Table 3. Recognition rates (%) with all the texture descriptors used here and in [4]

Texture descriptor	Number of zones	Best result
GLCM [4]	5	70.78±2.69
LBP [4]	5	80.33±1.67
Gabor filters	5	74.67±3.79
LPQ	1 (no zoning)	**80.78±0.77**

Table 3 presents the best results obtained with four different texture operators on the LMD. We have evaluated if there are statistically significant differences between these results. For this, the Friedman test with post hoc Shaffer's static procedure was employed. The multiple camparison statistical test has shown

that the p value of the statistical test was higher than the critical value in all cases at 95% confidence level. Thus, we have not found statistically significant difference between these results. This is favourable to LPQ, once it is the only one operator which presented the best result using global feature extraction.

Table 4 shows some results recently obtained on the LMD dataset using artist filter. Some of the works shown in this table refer to results presented in MIREX (Music Information Retrieval Evaluation eXchange) contest. In [15,16,17] the authors used acousitc features, extracted directly from the audio signal. One can see that the best result obtained here is among the best results.

Table 4. Best recognition rates (%) obtained on the LMD with artist filter

Work reference	Recognition rate (%)
Lopes et al. [14]	59.67±13.5
MIREX 2008 - LMD [15]	65.17±10.72
MIREX 2009 - LMD [16]	74.67±11.03
MIREX 2010 - LMD [17]	79.86±5.20
LBP (5 zones) [4]	80.33±1.67
LBP (Mel scale zoning) [5]	**82.33±1.45**
LPQ (this work)	**80.78±0.77**

On the one hand, one can say that the best recognition rate obtained on the LMD using visual features is that described in [5]. On the other hand, it is important to note that in that work a much bigger amount of classifiers (45) was created, since a nonlinear zoning with much more zones was used.

5 Conclusion

In this work we follow the investigation of the use of features extracted from the visual representation (spectrogram) of the audio signal in music genre recognition. We have compared the use of two different texture descriptors to capture the content of spectrogram images, i.e. Gabor filters and LPQ. We have tried two different approaches to deal with the intra-class variability of the spectrogram images, a global feature extraction and a feature extraction taking into account a linear zoning to obtain local information of the images.

The results obtained with LPQ texture operator are better than those obtained with Gabor filters. Regarding to results obtained with other texture descriptors on the LMD with global feature extraction or linear zoning, the result obtained with LPQ is the best one ever obtained. Interestingly, the global feature extraction performed slightly better than zoning with LPQ, unlike with Gabor filters and the other texture descriptor already investigated in other works.

In future works, we intend to develop experiments using LPQ descriptors with feature selection. The rationale behind this strategy is that one can reduce the dimensionality of the features vector and improve the performance either in terms of recognition rate or in terms of time.

Acknowledgments. The authors thank to CNPq (Grants 301653/2011-9, 402357 /2009-4), CAPES (Grant BEX5779/11-1) and Araucaria fundation (Grant 16767).

References

1. Tzanetakis, G., Cook, P.: Music Genre Classification of Audio Signals. IEEE Transactions on Speech and Audio Processing, 293–302 (2002)
2. Costa, Y.M.G., Oliveira, L.E.S., Koerich, A.L., Gouyon, F.: Music Genre Recognition Using Spectrograms. In: 18th International Conference on Systems, Signals and Image Processing. IEEE Press, Sarajevo (2011)
3. Haykin, S.: Advances in spectrum analysis and array processing, vol. 3. Prentice-Hall, NJ (1991)
4. Costa, Y.M.G., Oliveira, L.E.S., Koerich, A.L., Gouyon, F.: Comparing Textural Features for Music Genre Classification. In: WCCI 2012 IEEE World Congress on Computational Intelligence, Brisbane, Australia, pp. 1867–1872 (2012)
5. Costa, Y.M.G., Oliveira, L.E.S., Koerich, A.L., Gouyon, F., Martins, J.G.: Music genre classification using LBP textural features. Signal Processing 92(11), 2723–2737 (2012)
6. Kittler, J., Hatef, M., Duin, R.P., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 226–239 (1998)
7. Costa, C.H.L., Valle Jr., J.D., Koerich, A.L.: Automatic classification of audio data. In: IEEE International Conference on Systems, Man, and Cybernetics, pp. 562–567 (2004)
8. Silla Jr., C.N., Kaestner, C.A.A., Koerich, A.L.: Classificação de Gêneros Musicais Utilizando Vetores de Característica Híbridos. In: 13o Simpósio Brasileiro de Computação Musical (SBCM 2011), pp. 32–44 (2011)
9. Gabor, D.: Theory of communications. Journal of Institution of Electrical Engineers 93, 429–457 (1946)
10. Ojansivu, V., Heikkilä, J.: Blur insensitive texture classification using local phase quantization. In: Elmoataz, A., Lezoray, O., Nouboud, F., Mammass, D. (eds.) ICISP 2008 2008. LNCS, vol. 5099, pp. 236–243. Springer, Heidelberg (2008)
11. Vapnik, V.: The nature of statistical learning theory. Springer, New York (1995)
12. Silla Jr., C.N., Koerich, A.L., Kaestner, C.A.A.: The latin music database. In: Proceedings of the 9th International Conference on Music Information Retrieval, Philadelphia, USA, pp. 451–456 (2008)
13. Flexer, A.: A closer look on artist filter for musical genre classification. In: International Conference on Music Information Retrieval, pp. 341–344 (2007)
14. Lopes, M., Gouyon, F., Koerich, A.L., Oliveira, L.E.S.: Selection of Training Instances for Music Genre Classification. In: ICPR 2010 - 20th International Conference on Pattern Recognition, Istanbul, Turkey (2010)
15. MIREX: Music information retrieval evaluation exchange (2008), `http://www.music-ir.org/mirex/wiki/2008:Main_Page`
16. MIREX: Music information retrieval evaluation exchange (2009), `http://www.music-ir.org/mirex/wiki/2009:Main_Page`
17. MIREX: Music information retrieval evaluation exchange (2010), `http://www.music-ir.org/mirex/wiki/2010:Main_Page`