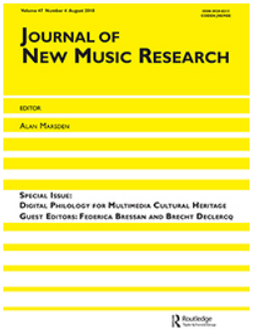

Journal of New Music Research

ISSN: 0929-8215 (Print) 1744-5027 (Online) Journal homepage: https://www.tandfonline.com/loi/nnmr20

Ensemble of deep learning, visual and acoustic features for music genre classification

Loris Nanni, Yandre M. G. Costa, Rafael L. Aguiar, Carlos N. Silla Jr. & Sheryl Brahnam

To cite this article: Loris Nanni, Yandre M. G. Costa, Rafael L. Aguiar, Carlos N. Silla Jr. & Sheryl Brahnam (2018) Ensemble of deep learning, visual and acoustic features for music genre classification, Journal of New Music Research, 47:4, 383-397, DOI: 10.1080/09298215.2018.1438476

To link to this article: https://doi.org/10.1080/09298215.2018.1438476

Published online: 21 Mar 2018.

JOURNAL OF NEW MUSIC RESEARCH
2018, VOL. 47, NO. 4, 383–397
https://doi.org/10.1080/09298215.2018.1438476

Routledge
Taylor & Francis Group

Ensemble of deep learning, visual and acoustic features for music genre classification

Loris Nanni[a], Yandre M. G. Costa[b], Rafael L. Aguiar[b,c], Carlos N. Silla Jr.[c] and Sheryl Brahnam[d]

[a]University of Padua, Italy; [b]State University of Maringa (UEM), Brazil; [c]PPGIa, Pontifical Catholic University of Paraná (PUCPR), Brazil; [d]Missouri State University, Missouri, USA

ABSTRACT

In this work, we present an ensemble for automated music genre classification that fuses acoustic and visual (both handcrafted and nonhandcrafted) features extracted from audio files. These features are evaluated, compared and fused in a final ensemble shown to produce better classification accuracy than other state-of-the-art approaches on the Latin Music Database, ISMIR 2004, and the GTZAN genre collection. To the best of our knowledge, this paper reports the largest test comparing the combination of different descriptors (including a wavelet convolutional scattering network, which has been tested here for the first time as an input for texture descriptors) and different matrix representations. Superior performance is obtained without ad hoc parameter optimisation; that is to say, the same ensemble of classifiers and parameter settings are used on all tested data-sets. To demonstrate generalisability, our approach is also assessed on the tasks of bird species recognition using vocalisation and whale detection data-sets. All MATLAB source code is available.

ARTICLE HISTORY
Received 12 October 2017
Accepted 31 January 2018

KEYWORDS
Audio classification; texture; image processing; acoustic features; ensemble of classifiers; machine learning

1. Introduction

Tzanetakis and Cook (2002) were probably the first to assess music genre classification as a pattern recognition problem. The authors introduced a comprehensive set of features designed to describe audio that were extracted from some common attributes typically thought to distinguish musical content, such as timbre, rhythm and pitch. In addition, the authors provided the research community with its first major music genre data-set, GTZAN, which provides 30 s samples of 1000 music pieces categorised into 10 different genres (classical, country, disco, hip hop, jazz, rock, blues, reggae, pop and metal), each of which contains 100 songs, making it a well-balanced data-set. Two years later, the International Society of Music Information Retrieval (ISMIR) released the ISMIR 2004 data-set composed of 1458 music pieces taken from six musical genres of music popular around the world (Gomez et al., 2006). This data-set was selected for the 'genre classification task' proposed as part of the Music Information Retrieval (MIR) contest organised by ISMIR and has since generated much interest in genre classification. Along with GTZAN, it has become a widely used benchmark in music genre classification. In 2008, Silla, Koerich, and Kaestner (2008) created the Latin Music Data-set (LMD). This data-set is composed of 3227 music recordings grouped into 10 Latin American genres categorised by a committee of professional ballroom dancing teachers who made their classifications based on their perception of how people would most likely dance to each work. The LMD data-set is even more challenging and fine-grained in its discriminations than other music genre data-sets since it is composed of ten Latin American musical genres (other datasets would classify the entire data-set simply as *Latin*).

These music genre data-sets and challenges have attracted the attention of a growing number of researchers in the machine learning community. Of particular interest has been the investigation of music descriptors that are based less on classical notions of musical features (such as timbre, rhythm and pitch, mentioned above) and more on the information extracted from some state-of-the-art descriptors that have already proven powerful in other machine learning classification problems.

In machine learning, there are broadly two types of descriptors in general use: those that are based on handcrafted features and those that are based on nonhandcrafted or automatically discovered features. Handcrafted features are designed and selected by human beings to

CONTACT Sheryl Brahnam sbrahnam@missouristate.edu Computer Information Systems, Missouri State University, 901 S. National, Springfield 65804, USA

extract specific characteristics from samples, and they tend to produce classifiers that are strongly dependent on feature engineering (Bengio, Courville, & Vincent, 2013; LeCun, Bengio, & Hinton, 2015). In contrast to handcrafted descriptors are methods that automatically discover and extract discriminative information from samples in a data-set. Along this line, some representation learning methods have been presented in the literature. Among the most common are deep learning methods, which have recently become popular thanks to the accessibility of Graphic Processing Units (GPUs), which have a massively parallel architecture specifically designed for handling multiple tasks simultaneously; GPUs greatly decrease the computational costs involved in training deep learners.

The Convolutional Neural Network (CNN) is a particularly powerful deep learning method that was introduced by LeCun et al. (1989). As far as we know, the first works that addressed Music Information Retrieval (MIR) tasks using CNN began in 2012. Humphrey and Bello (2012), for example, reviewed deep architectures and feature learning methods to discover alternative approaches suitable for solving challenges in the MIR research community. In addition, Humphrey, Bello, and LeCun (2012) assessed automatic chord detection and recognition using CNN, claiming that they achieved state-of-the-art performance in their first application of this method. Nakashika, Garcia, and Takiguchi (2012) performed experiments on the GTZAN data-set with good results using a CNN to classify feature maps obtained with the Gray Level Co-occurrence Matrix (GLCM) (Haralick, Shanmugam, & Dinstein, 1973) applied to a short-term mel spectrogram. A year later, Schlüter and Böck (2013) used CNN to perform musical onset detection. The authors observed that, although CNN slightly overcame the state-of-the-art at this task, CNN required less manual preprocessing. Gwardys and Grzywczak (2014) used a CNN model trained on a data-set composed of more than one million images labelled according to 1000 classes taken from Large Scale Visual Recognition Challenge (ILSVRC) 2012 edition; the trained model was then applied to spectrograms obtained from GTZAN music pieces. Before generating the spectrograms, a harmonic/percussive sound separation process was performed. The authors argued that the results obtained using spectrograms taken from the original sound, harmonic content and percussive content was close to the state-of-the-art. Finally, considering that, in many cases, the training time of CNN can become prohibitive on large datasets, Sigtia and Dixon (2014) focused on developing methods for overcoming this problem. The authors performed experiments on the GTZAN and ISMIR 2004 data-sets showing that by properly adjusting CNN parameters good performances can be obtained while significantly reducing the training time.

Classifying music using images, such as the spectrograms mentioned in some of the CNN studies above, is a fairly recent development despite the fact that spectrograms and other visual representations of sound have long proven valuable in sound analysis. In 2011, Costa, Oliveira, Koerich, and Gouyon (2011) started using texture features, specifically GLCM, extracted from spectrogram images, as did Wu and Jang (Wu et al., 2011), who also began combining visual and acoustic features for music genre classification. This initial research into visual images of music was quickly followed by studies that explored more powerful state-of-the-art texture descriptors, such as Local Binary Patterns (LBP), Local Phase Quantisation (LPQ) and Gabor filters (Costa, Oliveira, Koerich, & Gouyon, 2013; Costa, Oliveira, Koerich, Gouyon, & Martins, 2012), as well as ensembles of texture features extracted from spectrograms that were then combined with acoustic features (Nanni, Costa, Lucio, Silla, & Brahnam, 2016). In Costa, Oliveira, Koerich, Gouyon and Martins (2012) a method for splitting the spectrogram image into zones corresponding to frequency bands was introduced that had a positive effect on performance. The power of using visual descriptors of sound is illustrated by the wide use of visual-spectrogram techniques in other audio classification problems, such as language identification (Montalvo, Costa, & Calvo, 2015), speaker identification (Kekre, Kulkarni, Gaikar, & Gupta, 2012), speech and signal processing (Lu & Haung, 2016; Wu & Jang, 2015), food intake recognition by means of a throat microphone (Kalantarian et al., 2014), environmental sound classification (Kabayashi & Ye, 2014), bird identification (Lucio & Costa, 2015; Nanni, Costa, et al., 2016), and whale detection and identification (Nanni, Costa, Lucio, Silla, & Brahnam, 2017).

In this work, we expand previous studies presented in (Costa, Oliveira, & Silla, 2017; Nanni, Costa, et al., 2016; Nanni et al., 2017). In Costa et al. (2017), the authors started to investigate the complementarity between handcrafted features and CNN features in music classification tasks. In Nanni et al. (2017), Nanni, Costa, et al. (2016) the authors addressed the music genre classification task by combining handcrafted features obtained in the visual domain (spectrograms) with other features obtained directly from the audio signal.

In this paper, we explore the following types of audio images:

- Different spectrograms: three spectrogram images are created with the lower limits of the amplitudes set to –70 dBFS, –90 dBFS, and –120 dBFS, respectively.

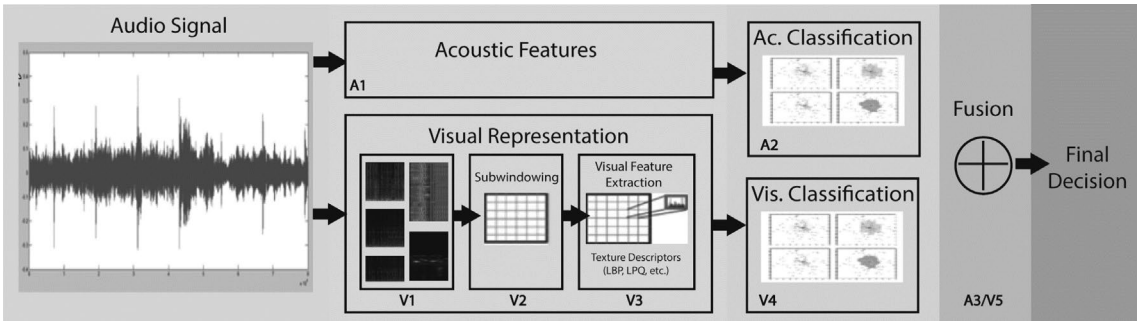

Figure 1. Visual and acoustic feature extraction and classification steps.

- Harmonic and Percussion Images: these images are generated using the Harmonic Percussion Separation (HPSS) Method proposed by Fitzgerald (2010).
- Scattergram: an image built from the ScatNet scattering representation (Bruna & Mallat, 2013).
- The Constant-Q transform: a method that produces the time-frequency representation image (Rakotomamonjy & Gasso, 2015).

Each image derived from the audio file is divided into a set of three subwindows from which a set of descriptors are extracted and trained on a classifier. The set of classifiers are then combined by sum rule.

Aside from exploring sets of descriptors taken from multiple images of music, some additional contributions of this work include the following:

- A dissimilarity space for audio classification based on the Shazam anchor point method is proposed;
- Exhaustive tests on the fusion between an ensemble of handcrafted descriptors and a system based on CNN is presented.
- For the first time, a wavelet convolutional scattering network is tested as input for texture descriptors.
- The MATLAB code used in all our experiments is freely provided to researchers for future comparisons (https://www.dropbox.com/s/bguw035yrqz0pwp/ElencoCode.docx?dl = 0).

Extensive experiments combining multiple acoustic descriptors with descriptors extracted from different audio images are carried out on the GTZAN, ISMIR 2004 and LMD benchmark databases. These experiments were designed to compare and maximise the performance obtained by varying combinations of descriptors. To the best of our knowledge, we report the largest comparisons of various combinations of different descriptors and different matrix representations. Experimental results show that our proposed system outperforms previous state-of-the-art approaches based on texture descriptors. Moreover, when handcrafted visual features are combined with CNN-based features along with acoustic features, the performance of the resulting system is better than other state-of-the-art approaches.

The remainder of this paper is organised as follows. In Section 2, we provide an overview of our proposed system, in Section 3 a description of the five types of audio images used in our approach, including details regarding the handcrafted and nonhandcrafted (CNN) descriptors extracted for these images, in Section 4 a discussion of the acoustic features extracted from the audio files, and in Section 5 a description of the general-purpose classifier used throughout our approach in all ensembles. In Section 6, we describe the GTZAN, ISMIR 2004, and LMD music genre data-sets in more detail and present the experimental results using these data-sets, including the results using two other data-sets representing two different sound domains. We do this to show the generalisability and power of our proposed system. Finally, we conclude this paper by summarising the main ideas behind our approach and by indicating our plan for the next phase of experimentation.

2. Proposed Approach

Our approach is schematised in Figure 1, illustrating how an audio signal is represented by two general types of features: acoustic features (A1) and audio image features (V1–3); these feature are classified using a separate Support Vector Machine (SVM) (A2 and V4), with results combined (A3 and V5) for a final ensemble decision.

As noted in Figure 1, visual feature extraction is accomplished in three steps:

Step V1: The original audio signal is transformed into five types of audio images: (1) Spectrogram, (2) Percussion, (3) Harmonic images, (4) Scattergram and (5) constant-Q transform (see Figure 2).
Step V2: Each image is divided into a set of subwindows.
Step V3: State-of-the-art local texture descriptors (described in detail in Section 3) are extracted from the subwindows, and each descriptor is classified using a separate SVM. In addition, a CNN is trained using the Spectrogram image.

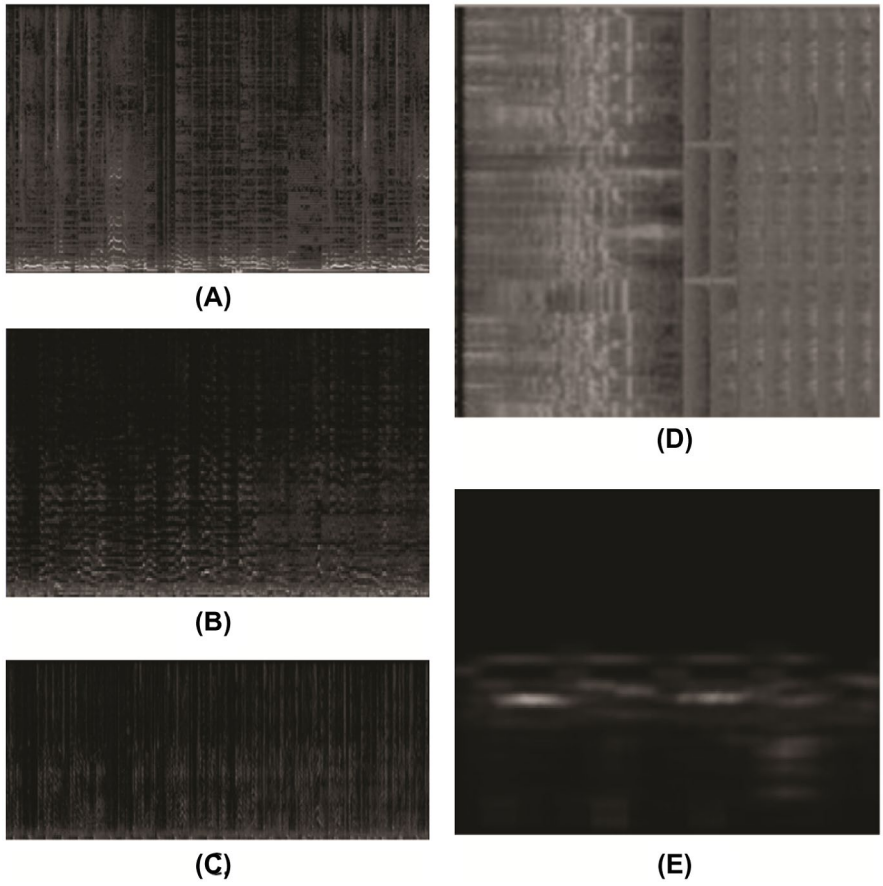

Figure 2. Five types of audio images: (a) spectrogram, (b) harmonic, (c) percussive (d) scatnet and (e) constant-Q transform.

Different types of acoustic features (described in detail in Section 4) are likewise extracted from the audio signal (A1), and each of these are classified using an SVM. The final decision of the ensemble (A3 plus V5) combines the audio visual scores obtained in Step V3 with the acoustic features (A2) using the weighed sum rule (described in Section 5 along with SVM).

3. Audio Image Representation

3.1. Step V1

In step V1 song files are transformed into five audio images, as illustrated in Figure 1. Each of these audio images are described below.

3.1.1. Spectrogram Images

The sample audio signals are converted into a spectrogram image that shows the spectrum of frequencies (vertical axis) as they vary in time (horizontal axis). The intensity of each point in the image represents the signal's amplitude. Spectrograms are generated using the Hanning window function with the DFT computed with a window size of 1024 samples. The audio sample rate is 22,050 Hz. Since no important difference exists between the content of the left/right audio channels, only the right channel is used. These images are then subjected to a battery of tests for finding complementarity among the different representations. Results of these tests led us to select the following three values: –70 dBFS, –90 dBFS and –120 dBFS.

3.1.2. Harmonic and Percussion Images

These images are produced using the Harmonic Percussion Separation (HPSS) method proposed by Fitzgerald (2010). Since this method works using a median filter across successive frames of the spectrogram of the audio signal, two images can be generated. If median filtering is performed across the frequency bins, the percussive events are enhanced, and the harmonic components are suppressed. If median filtering is performed across the time axis, the percussive events are suppressed, and the harmonic components are enhanced. These median filtered spectrograms are applied to the original spectrogram as masks to separate the harmonic and percussive parts of the signal, thereby generating the Harmonic and Percussion Images. In this work, we used the Librosa (McAfee, 2015) implementation of the HPSS method.

3.1.3. Scattergram

The scattergram is built from the ScatNet scattering representation, which produces an image much like a spectrogram that is the visualisation of the second-order translation-invariant scattering transform of a 1-D signal. ScatNet is a wavelet convolutional scattering network (Bruna & Mallat, 2013; Sifre & Mallet, 2012) that has achieved state-of-the-art results in many image recognition and music genre recognition challenges. ScatNet resembles a CNN in the sense that the scattering transform is the set of all paths that an input signal x might take from layer to layer. The convolutional filters are predefined as wavelets requiring no learning. Each layer in ScatNet is the association of a linear filter bank Wop with a non-linear operator: the complex modulus. Each operator Wop $\{1 + m\}$, with m the maximal order of the scattering transform, performs two operations resulting in two outputs: (1) an energy averaging operation by means of a low-pass filter according to the largest scale, ϕ, and (2) energy scattering operations along all scales using band-pass filters ψ_j, with j the scale index.

In audio processing the linear operators are constant-Q filter banks, with two layers typically sufficient for capturing the majority of the energy in a signal with an averaging window less than one second. The scattering operators rely on a set of built-in 'wavelet factories' that are suited to specific classes of signals. Wavelets are built by dilating a mother wavelet ψ by a factor $2^{1/Q}$ for some quality factor Q to obtain the filter bank,

$$\psi_j(t) = 2^{-j/Q}\psi\left(2^{-j/Q}t\right), \tag{1}$$

where the mother wavelet ψ is chosen such that adjacent wavelets barely overlap in frequency.

The scattering coefficients are defined by

$$S_1 x(t, j_1) = \left|x \star \psi_{j_1}\right| \star \phi(t) \tag{2}$$

$$S_2 x(t, j_1, j_2) = x \star \psi_{j_1} \left|\star \psi_{j_1}\right| \star \phi(t), \text{ and so on.}$$

The scattering representation S is a cell array, whose elements correspond to respective layers in the scattering transform.

In this study, we use the MATLAB tookbox ScatNet to generate the audio scattergrams.[1]

3.1.4. Constant-Q Transform

This method proposed in Rakotomamonjy and Gasso (2015), is a global feature extraction scheme that utilises information from a time-frequency representation (TFR) of an audio signal processed in such a way that it attenuates noise in the high-energy time-frequency structures. An audio signal is represented according to a constant-Q transform (Brown, 1991) (CQT), which produces the TFR image. The TFR image is then resized to a fixed size performed on the CQT matrix via a bicubic interpolation and is smoothed using mean filtering. Images are resized to 512 × 512.

3.2. Texture Descriptors

Because an audio signal is transformed into an image in our proposed method, image descriptors are used to analyse image similarities. Below we describe Local Binary Patterns (LBP), variants of LBP, specifically focusing on Local Phase Quantisation (LPQ) and its variants. We then describe Gabor filter feature extraction (GF), Binarised Statistical Image Features (BSIF), the LETRIST histogram (LEN) and the CodebookLess Model (CLM).

3.2.1. LBP

Many of the texture descriptors used in this paper are variants of LBP, a descriptor that has achieved great success due to its computational efficiency and discriminative power. Traditional LBP (Ojala, Pietikainen, & Maenpaa, 2002) can be expressed as

$$\mathrm{LBP}_{P,R} = \sum_{p=0}^{P-1} s(x)2^P, \tag{3}$$

where $x = q_{p-}q_c$ is the difference between the intensity levels of a central pixel (q_c) and a set of neighbouring pixels (q_p). A neighbourhood is defined by a circular region of radius R and P neighbouring points. The function $s(x)$ in Equation (3) is defined as:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

LBP codes range in $[0, 2^{P-1}]$, and LBP descriptors are the histograms of these binary numbers. The final descriptor is obtained by concatenating patterns at different radii R and different sampling points P. In our experiments: $(R = 1, P = 8)$ and $(R = 2, P = 16)$.

3.2.2. LBP Variants (LBP-HF, ELHF, RICLBP, AHP, HASC)

Many LBP variants have been proposed in the past ten years that aim at addressing some critical limitations of LBP: its sensitivity to rotation, blur and noise.

To achieve rotation invariance, Ahonen, Matas, He, and Pietikäinen (2009) proposed an effective LBP Histogram Fourier (LBP-HF) features that are computed from discrete Fourier transforms of LBP histograms. The LBP-HF

[1]This toolbox is available at http://www.di.ens.fr/data/software/scatnet/.

descriptor is formed first by computing a noninvariant LBP histogram over the whole region and then by constructing rotationally invariant features from the histogram. In our experiments, the multiscale LBP histogram Fourier descriptor is obtained from the concatenation of LBP-HF with values ($R = 1$, $P = 8$) and ($R = 2$, $P = 16$).

ELHF (Nanni, Brahnam, & Lumini, 2011) is an Ensemble of variants of the LHF that is built from the following seven descriptors, each trained by an SVM (with scores summed and normalised by dividing the sum by seven):

- FF: the original method, where from each Discrete Fourier transform (DFT) the first half of the coefficients are retained;
- DC: an approach where from each discrete Cosine transform (DCT) the first half of the coefficients are retained;
- An approach where the histogram is decomposed by Daubechies wavelet before DFT; then the method FF is performed;
- An approach where the histogram is decomposed by Daubechies wavelet before DCT; then the method DC is performed;
- An approach where the histogram is decomposed by Daubechies wavelet before DFT; then the method FF is performed with all coefficients retained;
- An approach where the histogram is decomposed by Daubechies wavelet before DCT; then the method DC is performed with all coefficients retained.
- An approach that retains all the bins of the histogram.

The Rotation Invariant Co-occurrence among adjacent LBPs (RICLBP), proposed by Nosaka, Suryanto, and Fukui (2012), is yet another rotation invariant descriptor that has high descriptive power. RICLBP introduces the concept of co-occurrence among LBPs in order to extract information related to the more global structures of the input image. In this paper we use RICLBP with radius: 1, 2 and 4.

Adaptive Hybrid Pattern (AHP), proposed by Zhu et al. (2015) is an LBP variant that is noise robust because a quantisation algorithm is applied that uses an equal probability quantisation to maximise partition entropy. AHP is thus robust to the impulsive noise in spectrograms and scattergrams, such as the background noise that emerge as spectral (horizontal) lines. The vector quantisation thresholds of AHP are adaptive to the content of the local patch with little discriminant information loss. In our experiments, quantisation_level = 5, and ($P = 8$, $R = 1$); ($P = 16$, $R = 2$).

The Heterogeneous Auto-Similarities of Characteristics (HASC) descriptor, proposed by San Biagio, Crocco, Cristani, Martelli, and Murino (2013) is applied to heterogeneous dense features maps. It encodes linear relations by covariances (COV) and non-linear associations through information-theoretic measures, i.e. entropy combined with mutual information (EMI). The combination of COV with EMI captures different features of the joint underlying probability density Functions (PDFs). This combination has been shown to boosted discriminative performance.

3.2.3. LPQ and Variants (Multiscale LPQ and Multiple LPQ, or MLPQ)

Local Phase Quantisation (LPQ) (Chan, Tahir, Kittler, & Pietikainen, 2013; Ojansivu & Heikkila, 2008) is an LBP variant that extracts the phase information in the frequency domain so that it is robust to blur variation. The local phase information is extracted using a 2D short-term Fourier transform (STFT) on a local window surrounding each pixel position. In the Fourier domain, the model for spatially invariant blurring of an image, $g(\mathbf{x})$, is

$$G(\mathbf{u}) = F(\mathbf{u})\,H(\mathbf{u}), \tag{5}$$

where $G(\mathbf{u})$, $F(\mathbf{u})$ and $H(\mathbf{u})$ are the discrete Fourier transforms (DFT) of the blurred image $g(\mathbf{x})$, the original image fx, and the point spread function (PSF) $h(\mathbf{x})$, respectively, and $\mathbf{u}$ is a vector of coordinates $[u, v]^{\mathrm{T}}$.

The magnitude and phase aspects in Equation (5) can be separated:

$$|G(\mathbf{u})| = |F(\mathbf{u})|\,|H(\mathbf{u})| \text{ and}$$
$$G(\mathbf{u}) = F(\mathbf{u}) + H(\mathbf{u}),$$

In the case where the blur $h(\mathbf{x})$ is centrally symmetric, the Fourier transform is always real-valued. Its phase is a two-valued function given by $\angle H(\mathbf{u}) = 0$ if $H(\mathbf{u}) \geq 0$ and π otherwise.

The LPQ method uses the local phase information extracted by STFT computed over the rectangular window/neighbourhood Nx of size M by M at each pixel position x of the image $f(\boldsymbol{x})$:

$$F(\boldsymbol{u}, \boldsymbol{x}) = \sum_{y \in Nx} f(x - y)\, e^{-j2\pi u^T y} = w_u^T fx \tag{7}$$

where w_u is the basis vector of the 2-D DFT at frequency $\mathbf{u}$, and fx is a vector containing all M^2 image samples from Nx.

Only four complex coefficients are considered. They correspond to the 2-D frequencies $\mathbf{u}_1 = [a,0]^{\mathrm{T}}$, $\mathbf{u}_2 = [0, a]^{\mathrm{T}}$, $\mathbf{u}_3 = [a, a]^{\mathrm{T}}$, and $\mathbf{u}_1 = [a,-a]^{\mathrm{T}}$, where a is the first frequency below the first zero crossing of $H(\mathbf{u})$ that satisfies $\angle G(\mathbf{u}) = \angle F(\mathbf{u})$ for all $\angle H(\mathbf{u}) \geq 0$.

If we let

$$F_x^c = [F(\boldsymbol{u}_1, \boldsymbol{x}), F(\boldsymbol{u}_2, \boldsymbol{x}), F(\boldsymbol{u}_3, \boldsymbol{x}), F(\boldsymbol{u}_4, \boldsymbol{x})], \text{ and} \tag{8}$$

$$Fx = [Re\{F_x^c\}, Im\{F_x^c\}]^T,$$

where Re{·} and Im{·} return the real and the imaginary parts of a complex number, respectively. The corresponding eight by M^2 transform matrix is

$$W = [Re\{w_{u1}, w_{u2}, w_{u3}, w_{u4}, \}, Im\{w_{u1}, w_{u2}, w_{u3}, w_{u4}, \}]^T \quad (9)$$

Thus,

$$Fx = \mathbf{W}fx. \quad (10)$$

Assuming that for fx the correlation coefficient between adjacent pixel values is ρ and the variance of each sample is $\sigma^2 = 1$, the covariance between positions x_i and x_j becomes $\sigma_{ij} = \rho^{||x_i - x_j||}$, where $||\cdot||$ denotes the L_2 norm.

The covariance matrix of the transform coefficient vector Fx can be obtained from $\mathbf{D} = \mathbf{WCW}^T$. Where $\mathbf{C}$ is the covariance matrix of all M samples in Nx.

The coefficients need to be decorrelated before quantisation. Assuming a Gaussian distribution, a whitening transform is applied:

$$\boldsymbol{Gx} = \mathbf{v}^T fx, \quad (11)$$

where $\mathbf{V}$ is an orthonormal matrix derived from the singular value decomposition (SVD) of the matrix $\mathbf{D}$, and $\mathbf{G}x$ is computed for all image positions.

The resulting vectors are quantised using a scalar quantiser, g_j is the jth component of $\mathbf{G}x$ and $q_j = 1$ if $g_j \geq 0$ and 0 otherwise. These quantised coefficients are represented as integers between 0 and 255 using the binary coding $b = \sum_{j=1}^{8} 2^{j-1}$. Finally, a histogram of these integer values is composed and used as a feature vector.

Multiscale LPQ is an LPQ variant that is computed regionally and adopts a component-based framework to maximise the insensitivity to misalignment, a phenomenon frequently encountered in blurring. Regional features are combined using kernel fusion.

Multiple LPQ is an ensemble of LPQ that was developed by Nanni, Brahnam, Lumini, and Barrier (2014), where different configurations of each LPQ trains a different classifier. The classifier results are combined by sum rule. MLPQ are examined by varying the following parameters: r (the neighbourhood size, where $r \in [1, 3, 5]$), a (the scalar frequency, where $a \in [0.8, 1, 1.2, 1.4, 1.6]$), and ρ (the correlation coefficient between adjacent pixel values, where $\rho \in [0.75, 0.95, 1.15, 1.35, 1.55, 1.75, 1.95]$). This is the same set proposed by Nanni et al. (2014) that avoids data overfitting. In addition, we evaluate another modified version of LPQ that was proposed in the same work that uses a ternary encoding scheme. To avoid the curse of dimensionality, each extracted descriptor is used to train a different SVM. This ensemble is built with 105 descriptors whose scores are summed and normalised by dividing the sum by 105.

3.2.4. BSIF

First proposed by Kannala and Rahtu (2012), BSIF extracts the standard Binarised Statistical Image Features by projecting subwindows of the entire image onto subspaces.

The canonical BSIF descriptor consists in assigning an n-bit label to each pixel of an image using a set (n) linear filters. This projects local image patches (size $l \times l$ pixels) onto a subspace. The n-bit label can be determined by binarisation:

$$s = \boldsymbol{Wx},$$

where $\boldsymbol{x}$ is the $l^2 \times 1$ vector notation of the $l \times l$ neighbourhood and $\boldsymbol{W}$ is a $n \times l^2$ matrix containing the compilation of the filters' vector notations. Specifically, the i-th digit of s is a function of the i-th linear filter $\boldsymbol{wi}$, and it is expressed as

$$s_i = \boldsymbol{w}_i^T \boldsymbol{x}. \quad (5)$$

Each bit of the *Bsif* code can be obtained as

$$b_i = \begin{cases} 1, & \text{if } s_i > 0 \\ 0, & \text{if } s_i \leq 0 \end{cases} \quad (6)$$

The set of filters $\boldsymbol{w}_i$ is created by maximising the statistical independence of the filter responses $\boldsymbol{si}$ on a set of patches from natural images by independent component analysis (Kannala & Rahtu, 2012).

The images are binarised using some threshold th. A histogram is built by maximising the statistical independence of the filter responses on a set of subwindows extracted from natural images by Independent Component Analysis (ICA) (see Kannala & Rahtu, 2012 for details). To increase this descriptor's discriminative power, an ensemble, as in Nanni, Paci, et al. (2016), is built by varying the following parameters of the approach: the filter size (size $\in$ {3, 5, 7, 9, 11}) and the threshold used for binarising ($th \in$ {–9, –6, –3, 0, 3, 6, 9}). The ensemble is built with thirty-five SVMs that are combined by sum rule. Each SVM is trained using a different feature vector extracted with a possible (*size*, *th*) combination of BSIF parameters. In the experimental section, we label this ensemble FullF.

3.2.5. GF

Gabor features (GF) extract global information about the frequency and orientation representations of the image. Gabor features are orientation and scale-tunable edge and line detectors, and the statistics of these microfeatures

Table 1. Parameters of ensemble of CLM.

Raw_feature	Reduction	Redim
eL2EMG	LRSMV	450
eSIFT	PCA	64
eSIFT	LRSVM	64

within a given region are used to characterise the underlying texture information.

Several different values for scale level and orientation are experimentally evaluated in Gabor. The best result was obtained using 5 different scale levels and 14 different orientations. The mean-squared energy and the mean amplitude were calculated from each possible combination between orientation and scale. This method produces a feature vector of size $5 \times 14 \times 2$.

3.2.6. LEN

LEN, proposed by Song and Meng (2017), explicitly encodes the joint information within an image across feature and scale spaces. LEN is a two-step process. In step 1, a set of transform features is constructed by applying linear and non-linear operators on the extremum responses of directional Gaussian derivative filters in scale space. In step 2, the scalar quantisation using binary or multilevel thresholding is adopted to quantise these transform features into texture codes. We use the default values available in the MATLAB toolbox.

3.2.7. CLM

The CLM proposed by Wang, Li, Zhang, and Zuoc (2016), is a dense sampling approach similar to Bag of Features (BoF). This approach differs from BoF in that it is not based on a codebook. Rather it represents the images with a single Gaussian model (see the original paper for details). In this work, we train three different CLM models as described in Table 1, which reports the type of extracted features (Raw_feature), the method for dimensionality reduction (Reduction), and the size of the reduced feature vector (Redim). The final score of the three CLMs models in Table 3 is produced via the sum rule.

3.3. Step V3: Convolutional Neural Networks

In this section, we describe each step that performs feature extraction and/or classification using CNN. As can be seen in Figure 3, a typical CNN consists of a set of layers each of which contains one or more planes. These planes receive input from a small neighbourhood in the planes of the previous layer and can be viewed as a feature map with a fixed feature detector that is convolved with a local window that is scanned over the planes in the previous layer. Multiple planes are typically used in each layer (called convolutional layers) so that multiple features can be detected. Once a feature has been detected, the exact location of that layer lessens in importance. For this reason, the convolutional layers are typically followed by another layer that performs a local averaging and subsampling operation. An example for a subsampling factor of two would be:

$$y_{ij} = \frac{x_{2i,2j} + x_{2i} + 1,\ x_{2j} + x_{2i},\ 2j + 1 + x_{2i} + 1.\ 2j + 1}{4} \tag{7}$$

where y_{ij} is the output of the same plane in the previous layer.

The network is trained using backpropagation gradient-descent (Haykin, 1994), and a connection strategy is used to reduce the number of weights in the network.

Taking into account that high resolution images are not suitable using CNN because of time constraints, we first downsize the images such that every four pixels is replaced by a single pixel, which reduces both image height and width by half. This is accomplished by taking only the first pixel of every four pixels in every 2×2 subwindow of the image. This strategy, inspired by Costa et al. (2017), is important to reduce the number of neurons in the convolutional layers and the number of trainable parameters in the first five layers of the network.

The CNN used here is composed of a 2D convolutional layer with 64 filters followed by a max-pooling layer. These steps are repeated twice. The 5th layer is a fully connected layer with 500 neurons. Up to this point the activation function used is always the rectified linear units (ReLUs). The number of neurons of the last layer must be equal to the number of classes of each problem and, in order to obtain class predictions for the samples, we use the activation function Softmax in this last layer. Training is performed using backpropagation with 50 epochs. For feature extraction purposes, we use the outcome of the 5th layer, which results in a 500-dimensional vector image representation. Figure 3 illustrates the architecture of the deep neural network used in this work.

3.4. Step V3: Shazam Dissimilarity Space

We propose using the well-know Shazam approach for building a dissimilarity space (A. Wang, 2003). Each pattern of the data-set is compared with all n patterns belonging to the training set, and for each comparison a match value is obtained. For each pattern this produces n match values (one for each training pattern) that are used to build the feature vector that describes the audio file. Finally, as with the texture descriptors, the feature vectors are fed into an SVM classifier.

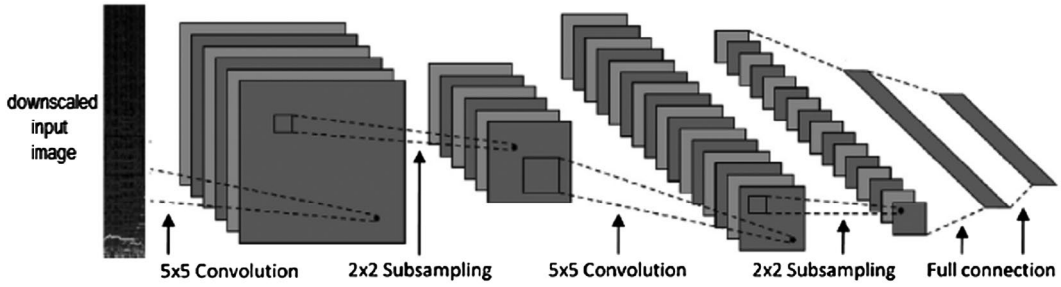

Figure 3. The deep Convolutional Neural Network architecture (adapted from Costa et al., 2017).

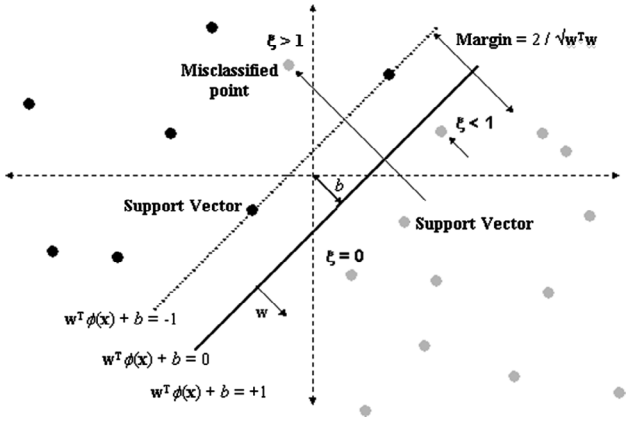

Figure 4. A graphical representation of the SVM hyperplane.

4. Acoustic Features (Step A1)

In this paper, we experiment and evaluate combining the following acoustic features in our system:

- Statistical Spectrum Descriptors (SSD) (Schroeder, Atal, & Hall, 1979), a set of statistical measures that describe audio content taken from the moments on the Sonogram (the Sone) of each of the twenty-four critical bands defined according to Bark scale.
- Rhythm Histogram (RH) (Schroeder et al., 1979), a feature set where the magnitudes of each modulation frequency bin of the twenty-four critical bands defined according to the Bark scale are summed up to form a histogram of 'rhythmic energy' per modulation frequency. The resulting histogram has 60 bins that reflect modulation frequencies between 0 and 10 Hz. The feature set is the median of the histograms of each 6s segment.
- Modulation Frequency Variance Descriptor (MVD) (Schroeder et al., 1979): a 420-dimensional feature vector that measures variation over the critical frequency bands for each modulation frequency. The MVD descriptor for the audio file is the mean of the MVDs taken from the 6s segments.
- Temporal Statistical Spectrum Descriptor (TSSD) (Fagerlund, 2007; Pachet & Zils, 2004), a feature set that incorporates temporal information from the SSD (timbre variations, changes in rhythm, etc.). Statistical measures are taken across the SSD measures extracted from segments at different time positions in an audio file.
- Temporal Rhythm Histograms (TRH) (Schroeder et al., 1979), a feature set that captures rhythmic changes in music over time.
- We also use the acoustic features of a commercial system for the music genre data-set. This system is based on a method proposed in (Lim, Lee, Jang, Lee, & Kim, 2012), which was later improved by Nanni, Costa, et al. (2016). The latter version is used in our experiments.

Each acoustic feature is trained on an SVM.

5. Classification and Fusion

The main classifier used in our ensemble is a one-versus-all SVM with a radial basis function (RBF) kernel. Introduced by Vapnik (1995), SVM is a learning system that is a highly effective general-purpose classifier that works well with ensembles (it is, in fact, the main classifier in many works using ensembles that are cited in this paper). Intuitively, SVM separates vectors representing two classes by finding an Optimal Separating Hyperplane (OSH) that separates the largest possible number of vectors belonging to the same class on the same side, while maximising the distance from either class to the hyperplane. The aim of OSH is to minimise the risk of misclassifying unseen samples in the testing set. In short, the basic two-class SVM formulation takes an input that is an implicit embedding Φ, and, with a labelled training set {xi}, it returns the hyperplane wT $\Phi(x) + b = 0$ that best separates the training samples of the two classes (see Figure 4).

SVM classifies patterns by applying different kernel functions $k(x, x')$ (e.g. linear, polynomial, RBF functions) as the possible sets of approximating functions. Different kernel functions are used depending upon the type of input patterns provided: a linear maximal margin classifier is used for linearly separable data, a linear soft

margin classifier is used for linearly nonseparable, or overlapping, classes, and a non-linear classifier, such as RBF, is used for classes that are overlapped as well as separated by non-linear hyperplanes.

The effectiveness of SVM depends on the kernel selected, the kernel's parameters, and soft margin parameter C, which is the regularisation parameter for the soft margin cost function of the SVM and controls the penalty for misclassifications for each support vector with the width of the separating margin: small C produces a large margin and large C a small margin.

The RBF kernel transforms the input space into a feature space of higher dimensionality, where a separating hyperplane is sought that separates the input vectors into two classes. Given labelled training data, $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{l}$, $x_i \in X \subset \mathcal{R}^d$, $y_i \in Y = \{-1, +1\}$, SVM constructs a maximal margin linear classifier in a high-dimensional feature space, $\Phi(x)$, defined by a positive definite kernel function specifying an inner product in the feature space. RBF can be defined as $k(x, x') = e^{-\gamma||x-x'||^2}$. The parameter γ in RBF is a free parameter that has an inverse relation to variance.

Since overfitting is a possibility when using small training sets, we perform no parameter optimisation; we simply set $C = 1000$ and $\gamma = 0.1$ for all experiments except for CLM, where a simple linear SVM with $C = 100$ is used. Before the classification step, the features are linearly normalised to [0, 1].

The outputs of any set of SVMs in an ensemble, whose results are combined by sum rule, with the final ensemble decision being the class that receives the largest support defined as

$$\sum_{cl=1}^{N} score(cl, j, x) \tag{8}$$

where N is the number of classifiers that belong to the ensemble, and score (cl,j,x) is the output x of the classifier cl with respect to a given class j.

6. Experimental Results

Using the recognition rate as the performance indicator, we assess the performance of our proposed approach on the following music genre datasets:

- LMD (Silla et al., 2008) is the Latin Music Database, which contains 3227 samples classified into ten musical genres: axe, bachata, bolero, forro, gaucha, merengue, pagode, salsa, sertaneja and tango. It was originally designed to evaluate music information retrieval systems. The testing protocol for this data-set is the threefold cross-validation protocol. The results reported here are the average recognition rate. The artist filter restriction (where all the samples by a specific artist are included in only one fold) is applied (Flexer, 2007). Since the distribution of samples per artist is not uniform, only a subset of 900 samples is used for fold creation.
- ISMIR 2004 (Gomez et al., 2006) is a genre classification data-set that contains 1458 samples assigned to six different genres: classical (640 samples), electronic (229), jazz/blues (52), metal/punk (90), rock/pop (203) and world (245). Because the number of music pieces per genre is not uniform, the artist filter restriction cannot be used. Moreover, it is not possible to include all the samples in the data-set since the signal segmentation strategy is used in this paper. As a result a total of 34 samples had to be excluded. Specifically, only 711 out of the 729 original samples are in the training set, and 713 out of the original 729 samples are in the testing set.
- GTZAN (Tzanetakis & Cook, 2002) contains 1000 music excerpts representing 10 genre classes: Blues, Classical, Country, Disco, Hip Hop, Jazz, Metal, Popular, Reggae and Rock. Each genre class contains 100 audio recordings that are 30s long. The results reported below refer to the average recognition rate obtained using the 10-fold cross-validation protocol, where the first 10 music pieces (0–9) of each genre are taken for Fold 1, and the next ten music pieces (10–19) from each genre are taken for Fold2, and so on. Unfortunately, GTZAN contains many replications, distortions and mislabelled samples (Sturm, 2012). We only include this data-set because it is still considered a benchmark for genre recognition. For a fair comparison of results reported in Lim et al. (2012), we also evaluate the performance using the same 10-fold split protocol tested and shared by the authors (Lim et al., 2012), a protocol which we label *GTZAN1* in the experimental results.

To demonstrate the power and generalisability of the proposed approach, we evaluate it on the following two animal sound benchmark data-sets:

- BIRD is the Bird Songs 46 data-set (Nanni, Costa, et al., 2016)[2] and was developed as a subset of that used in (Nanni, Costa, et al., 2016). To build the subset, all bird species with less than 10 samples were removed. As a result, the Bird Songs 46 data-set is composed of 2814 audio samples of bird vocalisation taken from 46 different species found in the South of Brazil. Although the Bird Songs 46

[2]This data-set is available at www.din.uem.br/yandre/birds/bird_songs_46.tar.gz.

Table 2. Comparison of tested descriptors (described above) using accuracy.

New descriptors	LMD	ISMIR	GTZAN
MLPQ_S	85.1	84.4	85.4
FullF_S	87.1	86.0	84.9
CLM_S	86.7	84.9	86.3
EasyO_S	86.3	85.5	85.5
EasyO_hp	89.0	84.4	85.6
EasyO_S + EasyO_hp	**90.0**	85.8	86.3
EasyO_S + EasyO_hp + Easy_scat	89.9	87.0	87.0
EasyO_S + EasyO_hp + Easy_scat + Easy_cqt	89.8	86.5	87.3
FUS	**90.0**	87.1	88.4
FUSnoHP	87.1	87.7	88.5
3 × FUS + 0.5 × CLMfus	89.3	87.8	89.4
CNN	**91.7**	87.1	77.0
CNN+H	**91.7**	**91.0**	**93.6**
ShD	32.4	48.5	30.6

Note: Bold values are the best classification scores.

data-set is composed exclusively of bird songs, in some cases calls related to other bird species can be heard in the background. The protocol used for this data-set is a stratified 10-fold cross validation strategy.

- WHALE is the whale identification data-set used in 'The Marinexplore and Cornell University Whale Detection Challenge'.[3] This data-set is composed of 84,503 audio clips that are 2s long and that contain mixtures of right whale calls, non-biological noise, and other whale calls. Thirty thousand samples have class labels. In this work, we used the first 20,000 samples as the training set and the remaining 10,000 samples for the testing set. The results on this data-set are described using the Area under the ROC Curve (AUC), which is the same performance indicator that was used in the whale detection challenge. AUC is a scalar measure ranging between 0 and 1 (best performance). It can be interpreted as the probability that the classifier will assign a higher score to a randomly picked positive sample than to a randomly picked negative sample (Qin, 2006).

In Table 2 the methods labelled K × A + K1 × B refer to a weighted sum rule between A (with weight K) and B (with weight K1). The methods labelled A_IMG, where IMG is either S (spectrograms), hp (harmonic-percussion), scat (scattergram), or cqt (constant q-transform), refers to the feature extraction method A that was performed on IMG.

We also report in Table 2 the performance of the following ensembles of descriptors:

- EasyO_IMG, which is the sum rule among LPQ, ELHF, LBP, RICLBP, HASC, LET and GF.
- CLMfus, which is the sum rule between CLM trained with spectrograms and CLM trained with the scattergram. Due to the computation time restraints, CLM is trained using only the scattergram and spectrograms.
- CNN is the fusion between CNN trained with spectrograms and harmonic/percussion images; this restriction also due to computation time.
- FUS is the weighted sum rule A + 0.5 × B, where, A = (EasyO_S + EasyO_hp + Easy_scat) and B = (MLPQ_S + 2 × FullF_S + MLPQ_hp + 2 × FullF_hp + MLPQ_scat + 2 × FullF_scat). Before fusion, the scores of A and B are normalised to mean 0 and standard deviation 1.
- FUSnoHP is the weighted sum rule A + 0.5 × B, where A = (EasyO_S + Easy_scat) and B = (MLPQ_S + 2 × FullF_S + MLPQ_scat + 2 × FullF_scat). Before fusion, the scores of A and B are normalised to mean 0 and standard deviation 1.
- CNN+H is the sum rule between CNN and FUS. Before fusion, the scores of A and B are normalised to mean 0 and standard deviation 1.

Since ShD performs poorly, it is not useful in fusion. Clearly results are not random. Shazam is able to build a feature vector that can represent the audio files, however poorly. More tests in the future should be performed to improve this approach (tests that examine, for instance, how best to select the prototype for building the dissimilarity space).

Due to the high computation time of CLM and considering as well the similar performance between FUS and 3 × FUS + 0.5 × CLMfus, our proposed ensemble of handcrafted features is the set labelled FUS.

In our opinion, the performance of CNN+H is very interesting. It performs better than both CNN and FUS; this experimental result convincingly demonstrates that CNN and handcrafted features extract different types of information from images. It is, therefore, useful to combine them.

In Table 3 we compare our proposed ensemble of handcrafted texture descriptors, FUS, with other approaches that have used texture descriptors for describing an audio pattern. For better assessing the performance of our method, in Table 3 we report its performance on three other datasets. The new set of descriptors proposed here clearly outperforms other similar approaches reported in the literature.

We also report in Table 3 the performance of the following:

- FUSnoMLPQ, which is FUS with the MLPQ descriptor discarded;
- FUSnoHP_noMLPQ, which is FUSnoHP with the MLPQ descriptor discarded.

[3]Available at www.kaggle.com/c/whale-detection-challenge.

Table 3. Comparison of ensemble of handcrafted descriptors using accuracy (WHALE uses AUC).

Descriptor	BIRD	BIRD2	WHALE	LMD	ISMIR	GTZAN
FUSnoHP	**90.2**	**97.4**	**94.0**	87.1	**87.7**	**88.5**
FUSnoHP_noMLPQ	89.9	97.0	93.5	86.0	87.4	88.2
FUS	89.5	95.2	**94.0**	**90.0**	87.1	88.4
FUSnoMLPQ	89.4	94.7	93.9	**90.0**	87.1	87.6
Nanni et al. (in press)	89.9	89.2	93.3	**90.0**	85.3	87.4
Nanni et al. (2017)	89.2	85.1	92.2	86.2	82.2	86.1
Nanni, Costa, Lumini, Kim, and Baek (2015)	85.9	83.1	87.1	86.1	81.6	83.8
Costa, Oliveira, Koerich, Gouyon and Martins (2012)				82.3	82.1	
Costa, Oliveira, Koerich, and Gouyon (2012)				70.7		
Wu et al. (2011)					82.2	82.1
Costa et al. (2013)					80.8	
Gwardys and Grzywczak (2014)						78.0

Note: See description of this data-set above.

Table 4. Comparison with the literature.

Work/Method	ISMIR 2004	GTZAN	LMD
HERE	**92.1**	**95.7**	**90.3**
Nanni et al. (in press)	90.9	90.8	86.2
Nanni et al. (2015)	90.2	89.8	85.1
Nanni et al. (2016)	90.2	89.9	86.1
Nanni et al. (2017)	90.9	90.6	84.6
Costa et al. (2017)	87.1	–	92.0
Senac, Pellegrini, Mouret, and Pinquier (2017)	–	91.0	–
Wu et al. (2011)	86.1	86.1	–
Tzanetakis and Cook (2002)	–	61.0	–
Lim et al. (2012)	89.9	87.4	–
Hamel (2011)	–	–	82.3
Ren and Jang (2012)	–	–	77.0
Pikrakis (2013)	–	–	77.6
Seyerlehner, Schedl, Pohle, and Knees (2010)	88.3	79.9	79.9
Panagakis, Kotropoulos, and Arce (2009)	85.5	89.4	–
Costa, Oliveira, Koerich, Gouyon, and Martins (2012)	80.65	–	82.33
Aguiar, Costa, and Nanni (2016)	–	–	88.56
Lopes, Gouyon, Koerich, and Oliveira (2010)	–	–	59.7
Cao and Li (2009)	82.1	79.0	74.7
Ren and Jang (2012)	–	–	77.0
Hamel (2011)	–	–	82.3
Marques, Lopes, Sordo, Langlois, and Gouyon (2010)	79.8		
Lidy, Rauber, Pertusa, and Inesta (2007)	81.4		
Sigtia and Dixon (2014)	74.4		

It is interesting to note that discarding the expensive MLPQ produces a minimal drop in performance. Moreover, in both the BIRD data-sets, HP images perform rather poorly, with FUSnoHP outperforming FUS in both BIRD and BIRD2: (note: in BIRD2, the performance of Easy_scat is 93.9%, the performance of Easy_S is 86.6%, and the performance of Easy_hp is 84.7. In BIRD, the performance of Easy_scat is 77.1%, the performance of Easy_S is 88.6%, and the performance of Easy_hp is 54.5%).

Nonetheless, compared to the literature, the proposed ensemble FUS works very well across all the tested data-sets.

In Table 4 we report the performance obtained by the ensembles that combine acoustic and visual features (labelled HERE), where the fusion by sum rule is between CNN+H, the visual features, and Ac, the acoustic features. Before the fusion, the scores of CNN+H and Ac are normalised to mean 0 and standard deviation 1. For Ac we used the same acoustic features that were tested in Nanni et al. (2017).

Results show that our proposed ensemble obtains state-of-the-art-performance in all the tested music genre datasets.

To further validate our ensemble we compare the different set of descriptors by Wilcoxon signed rank test (Demšar, 2006). FUS outperforms all previous published ensembles using visual features reported in Table 3 with a p-value of 0.1.

To evaluate the usefulness of the fusions between the different visual descriptors, we computed Yule's Q-statistic (Kuncheva & Whitaker, 2003), which checks the error independence of the different SVMs, each of which in our approach were trained with different descriptors. The values of Q lie between $[-1,1]$. Classifiers that erroneously classify the same patterns have $Q < 0$; those that correctly classify the same pattern have $Q > 0$. The average Q-statistic among the proposed ensembles of visual descriptors is 0.811. We also measured the average Q-statistic between the visual and acoustic features, obtaining 0.795. These values show that the SVMs trained with different descriptors capture different types of information. For this reason, their fusion improves the performance of the stand-alone approaches.

7. Conclusion and Future Work

In this work, we presented a new and effective ensemble for automated music genre classification that is based on the fusion of sets of acoustic and visual features extracted from audio files of songs. These features are then evaluated, compared, and fused. The visual features are taken from images that were constructed using different methods of representing an audio file as an image. These images are

divided into subwindows from which a set of local texture descriptors are extracted. For each texture descriptor a different Support Vector Machine (SVM) is trained, and the SVMs are summed for a final decision. Moreover, exhaustive tests on the fusion between the proposed ensemble of handcrafted texture descriptors and a system based on convolutional neural networks is reported. Finally, different acoustic features are evaluated. The experiments demonstrate that the fusion of different texture features result in state-of-the-art performance; however, not all fusions of texture features combine equally well with audio features to improve performance. Nonetheless, our proposed ensemble that combines texture features with audio features obtains results that are comparable with existing audio signal approaches.

In the future, we plan on adding other data-sets to those used in the experiments reported here in order to obtain more complete validation of the proposed ensemble. We also plan on testing this system with different sounds.

Finally, we want to highlight the fact that the approach based on the extraction of visual features is freely available to other researchers for future comparisons. MATLAB code will be located at https://www.dropbox.com/s/bguw035yrqz0pwp/ElencoCode.docx?dl = 0.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Aguiar, R. L., Costa, Y. M. G., & Nanni, L. (2016). *Music genre recognition using spectrograms with harmonic-percussive sound separation*. Paper presented at the 35th international conference of the Chilean computer science society, Valparaiso, Chile.

Ahonen, T., Matas, J., He, C., & Pietikäinen, M. (2009). Rotation invariant image description with local binary pattern histogram fourier features, Image Analysis, SCIA 2009. *Lecture Notes in Computer Science, 5575*, 61–70.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1798–1828.

Brown, J. (1991). Calculation of a constant Q spectral transform. *The Journal of the Acoustical Society of America, 89*(1), 425–434.

Bruna, J., & Mallat, S. (2013). Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1872–1886.

Cao, C., & Li, M. (2009). *Thinkit's Submission for MIREX 2009 Audio Music Classification and Similarity Tasks (MIREX-09)*. Paper presented at the MIREX abstracts, International Conference on Music Information, Kobe, Japan.

Chan, C., Tahir, M., Kittler, J., & Pietikainen, M. (2013). Multiscale local phase quantisation for robust component-based face recognition using kernel fusion of multiple descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(5), 1164–1177.

Costa, Y. M. G., Oliveira, L. S., Koerich, A. L., & Gouyon, F. (2011). *Music genre recognition using spectrograms*. Paper presented at the 18th International Conference on Systems, Signals and Image Processing, Sarajevo.

Costa, Y. M. G., Oliveira, L. S., Koerich, A. L., & Gouyon, F. (2012). *Comparing textural features for music genre classification*. Paper presented at the IEEE World Congress on Computational Intelligence, Brisbane.

Costa, Y. M. G., Oliveira, L. S., Koerich, A. L., & Gouyon, F. (2013). *Music genre recognition using gabor filters and LPQ texture descriptors*. Paper presented at the 18th Iberoamerican Congress on Pattern Recognition, Havana.

Costa, Y. M. G., Oliveira, L. S., Koerich, A. L., Gouyon, F., & Martins, J. (2012). Music genre classification using LBP textural features. *Signal Processing, 92*, 2723–2737.

Costa, Y. M. G., Oliveira, L. E. S., & Silla, C. N., Jr. (2017). An evaluation of convolutional neural networks for music classification using spectrograms. *Applied Soft Computing, 52*, 28–38.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Fagerlund, S. (2007). Bird species recognition using support vector machines. *EURASIP Journal on Applied Signal Processing, 2007*, 1–8. doi:10.1155/2007/38637.

Fitzgerald, D. (2010). *Harmonic/Percussive separation using median filtering*. Paper presented at the 13th International Conference on Digital Audio Effects (DAFx-10, Graz, Austria.

Flexer, A. (2007). A closer look on artist filters for musical genre classification. *World, 19*(122), 16–17.

Gomez, E., Gouyon, F., Herrera, P., Koppenberger, M., Ong, B., Serra, X., … Wack, N. (2006). *ISMIR 2004 audio description contest*. Barcelona, Spain.

Gwardys, G., & Grzywczak, D. (2014). Deep image features in music information. *International Journal of Electronics and Telecommunications, 60*(4), 321–326.

Hamel, P. (2011). *Pooled features classification. Paper presented at the Submission to Audio Train/Test Task of MIREX*. Retrieved from http://www.music-ir.org/mirex/abstracts/2011/PH1.pdf

Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics, 3*(6), 610–621.

Haykin, S. (1994). *Neural networks, a comprehensive foundation*. New York, NY: Macmillan.

Humphrey, E., & Bello, J. P. (2012). *Rethinking automatic chord recognition with convolutional neural networks. Paper presented at the International Conference on Machine Learning and Applications*, Boca Raton, FL.

Humphrey, E., Bello, J. P., & LeCun, Y. (2012). Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In *International Conference on Music Information* (pp. 403–408). Porto.

Kabayashi, T., & Ye, J. (2014). *Acoustic feature extraction by statistics based local binary pattern for environmental sound classification*. Paper presented at the IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP), Florence.

Kalantarian, H., Alshurafa, N., Pourhomayoun, M., Sarin, S., Le, T., & Sarrafzadeh, M. (2014). *Spectrogram-based audio classification of nutrition intake*. Paper presented at the Health Innovations and Point-of-Care Technologies Conference, Seattle, WA.

Kannala, J., & Rahtu, E. (2012). *Bsif: Binarized statistical image features*. Paper presented at the 21st International Conference on Pattern Recognition (ICPR 2012), Tsukuba, Japan.

Kekre, H. B., Kulkarni, V., Gaikar, P., & Gupta, N. (2012). Speaker identification using spectrograms of varying frame sizes. *International Journal of Computer Applications, 50*(20), 27–33.

Kuncheva, L. I., & Whitaker, C. J. (2003). Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning, 51*(2), 181–207.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation, 1*(4), 541–551.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Lidy, T., Rauber, A., Pertusa, A., & Inesta, J. M. (2007). *Improving genre classification by combination of audio and symbolic descriptors using a transcription system*. *Paper presented at the ISMIR*, Vienna, Austria.

Lim, S.-C., Lee, J.-S., Jang, S.-J., Lee, S.-P., & Kim, M. Y. (2012). Music-genre classification system based on spectro-temporal features and feature selection. *IEEE Transactions on Consumer Electronics, 58*(4), 1262–1268.

Lopes, M., Gouyon, F., Koerich, A., & Oliveira, L. E. S. (2010). *Selection of training instances for music genre classification*. Paper presented at the 20th International Conference on Pattern Recognition, Istanbul.

Lu, F., & Haung, J. (2016). *An improved local binary pattern operator for texture classification*. Paper presented at the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai.

Lucio, D. R., & Costa, Y. M. G. (2015). *Bird species classification using spectrograms*. Paper presented at the XLI Latin American Computing Conference (CLEI), Arequipa, Peru.

Marques, G., Lopes, M., Sordo, M., Langlois, T., & Gouyon, F. (2010). *Additional evidence that common low-level features of individual audio frames are not representative of music genres*. Paper presented at the Sound and Music Computing Conference, Barcelona, Spain.

McAfee, B. (2015). Librosa: Audio and music signal analysis in python. Paper presented at the 14th Python in Science Conference (SCIPY), Austin, TX.

Montalvo, A., Costa, Y. M. G., & Calvo, J. R. (2015). Language identification using spectrogram texture. *Progress in pattern recognition, image analysis, computer vision, and applications* (pp. 543–550). Berlin: Springer.

Nakashika, T., Garcia, C., & Takiguchi, T. (2012). Local-feature-map integration using convolutional neural networks for music genre classification. *Interspeech,* 1752–1755.

Nanni, L., Aguiar, R. L., Costa, Y. M .G., Brahnam, S., Silla, C. N., & Brattin, R. L. (in press). Bird and whale species identification using sound images. *IET Computer Vision*. doi:10.1049/iet-cvi.2017.0075.

Nanni, L., Brahnam, S., & Lumini, A. (2011). Combining different local binary pattern variants to boost performance. *Expert Systems with Applications, 38*(5), 6209–6216.

Nanni, L., Brahnam, S., Lumini, A., & Barrier, T. (2014). Ensemble of local phase quantization variants with ternary encoding. In S. Brahnam, L. C. Jain, A. Lumini, & L. Nanni (Eds.), *Local binary patterns: New variants and applications*. (pp. 177–188). Berlin: Springer-Verlag.

Nanni, L., Costa, Y., Lucio, D. R., Silla, C. N., Jr., & Brahnam, S. (2016). *Combining visual and acoustic features for bird species classification*. Paper presented at the IEEE International Conference on Tools with Artificial Intelligence (ICTAI), San Jose, CA (pp. 396–401).

Nanni, L., Costa, Y. M. G., Lucio, D. R., Silla, C. N., Jr., & Brahnam, S. (2017). Combining visual and acoustic features for audio classification tasks. *Pattern Recognition Letters, 88*(March), 49–56.

Nanni, L., Costa, Y. M. G., Lumini, A., Kim, M. Y., & Baek, S. R. (2015). Combining visual and acoustic features for music genre classification. *Expert Systems with Applications, 45*(1), 108–117.

Nanni, L., Paci, M., Santos, F. L. C. d., Skottman, H., Juuti-Uusitalo, K., & Hyttinen, J. (2016). Texture descriptors ensembles enable image-based classification of maturation of human stem cell-derived retinal pigmented epithelium. *PLoS ONE,* (e0149399).

Nosaka, R., Suryanto, C. H., & Fukui, K. (2012). *Rotation invariant co-occurrence among adjacent LBPs*. Paper presented at the ACCV Workshops, Daejeon.

Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(7), 971–987.

Ojansivu, V., & Heikkila, J. (2008). *Blur insensitive texture classification using local phase quantization*. Paper presented at the ICISP, Cherbourg-Octeville.

Pachet, F., & Zils, A. (2004). *Automatic extraction of music descriptors from acoustic signals*. Paper presented at the 5th International Conference on Music Information Retrieval (ISMIR), Barcelona, Spain.

Panagakis, Y., Kotropoulos, C., & Arce, G. R. (2009). *Music genre classification using locality preserving non-negative tensor factorization and sparse representations*. Paper presented at the 10th International Conference on Music Information, Kobe, Japan.

Pikrakis, A. (2013). Audio latin music genre classification: A MIREX submission based on a deep learning approach to rhythm modelling. In *14th International Society for Music Information Retrieval Conference*, Curitiba, Brazil.

Qin, Z. C. (2006). *ROC analysis for predictions made by probabilistic classifiers*. Paper presented at the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou.

Rakotomamonjy, A., & Gasso, G. (2015). Histogram of gradients of time–frequency representations for audio scene classification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23*(1), 142–153.

Ren, J.-M., & Jang, J.-S. R. (2012). Discovering time-constrained sequential patterns for music genre classification. *IEEE Transactions on Audio, Speech, and Language Processing, 20*(4), 1134–1144.

San Biagio, M., Crocco, M., Cristani, M., Martelli, S., & Murino, V. (2013). *Heterogeneous auto-similarities of characteristics (hasc): Exploiting relational information for classification*. Paper presented at the IEEE Computer Vision (ICCV13), Washington, DC.

Schlüter, J., & Böck, S. (2013). *Musical onset detection with convolutional neural networks*. Paper presented at the International Workshop on Machine Learning and Music, Prague, Czech Republic.

Schroeder, M. R., Atal, B. S., & Hall, J. L. (1979). Optimizing digital speech coders by exploiting masking properties of the human ear. *The Journal of the Acoustical Society of America, 66*(6), 1647–1652.

Senac, C., Pellegrini, T., Mouret, F., & Pinquier, J. (2017). *Music feature maps with convolutional neural networks for music genre classification*. *Paper presented at the 15th International Workshop on Content-Based Multimedia Indexing*, Florence, Italy.

Seyerlehner, K., Schedl, M., Pohle, T., & Knees, P. (2010). *Using block-level features for genre classification, tag classification and music similarity estimation*. Paper presented at the 6th Annual Music Information Retrieval Evaluation eXchange (MIREX-2010), Utrecht, Netherlands.

Sifre, L., & Mallet, S. (2012). *Combined scattering for rotation invariant texture analysis*. *Paper presented at the European Symposium Artificial Neural Networks*, Bruges.

Sigtia, S., & Dixon, S. (2014). *Improved music feature learning with deep neural networks*. *Paper presented at the IEEE International Conference on Acoustic, Speech and Signal Processing*, Florence.

Silla, C. N., Koerich, A. L., & Kaestner, C. A. A. (2008). *The latin music database*. *Paper presented at the 9th International Conference on Music Information*, Philadelphia.

Song, T., & Meng, F. (2017). Letrist: Locally encoded transform feature histogram for rotation-invariant texture classification. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99).

Sturm, B. L. (2012). *An analysis of the GTZAN music genre dataset*. Paper presented at the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies, Nara.

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing, 10*(5), 293–302.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York, NY: Springer-Verlag.

Wang, A. (2003). *An industrial strength audio search algorithm*. Paper presented at the ISMIR Proceedings, Baltimore, MD.

Wang, Q., Li, P., Zhang, L., & Zuoc, W. (2016). Towards effective codebookless model for image classification. *Pattern Recognition, 59*, 63–71.

Wu, M.-J., & Jang, J.-S. R. (2015). Combining acoustic and multilevel visual features for music genre classification. *ACM Transactions on Multimedia Computing Communications and Applications, 12*(1), 10:1–10:17.

Wu, M. J., Chen, Z. S., Jang, J. S. R., Ren, J. M., Li, Y. H., & Lu, C. H. (2011). Combining visual and acoustic feature for music genre classification. *Paper presented at the International Conference on Machine Learning and Applications*, Honolulu, HI.

Zhu, Z., You, X., Chen, C. L. P., Too, D., Ou, W., Jiang, X., & Zoe, J. (2015). An adaptive hybrid pattern for noise-robust texture analysis. *Pattern Recognition, 48*, 2592–2608.