

# Analytic Comparison of Audio Feature Sets using Self-Organising Maps

Rudolf Mayer, Jakob Frank, Andreas Rauber  
Institute of Software Technology and Interactive Systems  
Vienna University of Technology, Austria  
{mayer,frank,rauber}@ifs.tuwien.ac.at

**Abstract**—A wealth of different feature sets for analysing music has been proposed and employed in several different Music Information Retrieval applications. In many cases, the feature sets are compared with each other based on benchmarks in supervised machine learning, such as automatic genre classification. While this approach makes features comparable for specific tasks, it doesn't reveal much detail on the specific musical characteristics captured by the single feature sets. In this paper, we thus perform an analytic comparison of several different audio feature sets by means of Self-Organising Maps. They perform a projection from a high dimensional input space (the audio features) to a lower dimensional output space, often a two-dimensional map, while preserving the topological order of the input space. Comparing the stability of this projection allows to draw conclusions on the specific properties of the single feature sets.

## I. INTRODUCTION

One major precondition for many Music Information Retrieval (MIR) tasks is to adequately describe music, resp. its sound signal, by a set of (numerically processable) feature vectors. Thus, a range of different audio features has been developed, such as the Mel-frequency cepstral coefficients (MFCC), the set of features provided by the MARSYAS system, or the Rhythm Patterns, Rhythm Histograms and Statistical Spectrum Descriptors suite of features.

All these feature sets capture certain different characteristics of music, and thus might perform unequally well in different MIR tasks. Very often, feature sets are compared by the means of benchmarks, e.g. the automated classification of music towards a certain label, such as in automatic genre classification. While this allows a comparative evaluation of different feature sets with respect to specific tasks, it doesn't provide many insights on the properties of each feature set. On the other hand, clustering or projection methods can reveal information such as which data items tend to be organised together, revealing information on the acoustic similarities captured by the respective feature sets. Building on this assumption, we utilise a recently developed method to compare different instances of a specific projection and vector quantisation method, the Self-Organising Maps, to compare how the resulting map is influenced by the different feature sets.

The remainder of this paper is structured as follows. Section II discusses related work in Music Information Retrieval and Self-Organising Maps, while Section III presents the employed audio features in detail. Section IV then introduces the method for comparing Self-Organising Maps. In Section V we

introduce the dataset used, and discuss experimental results. Finally, Section VI gives conclusions and presents future work.

## II. RELATED WORK

Music Information Retrieval (MIR) is a discipline of Information Retrieval focussing on adequately describing and accessing (digital) audio. Important research directions include, but are not limited to, similarity retrieval, musical (genre) classification, or music analysis and knowledge representation.

The dominant method of processing audio files in MIR is by analysing the audio signal. A wealth of different descriptive features for the abstract representation of audio content have been presented. The feature sets we used in our experiments, i.e. *Rhythm Patterns* and derived sets, *MARSYAS*, and *Chroma*, are well known algorithms focusing on different audio characteristics and will be described briefly in Section III.

The Self-Organising Map (SOM) [1] is an artificial neural network used for data analysis in numerous applications. The SOM combines principles of vector projection (mapping) and vector quantisation (clustering), and thus provides a mapping from a high-dimensional input space to a lower dimensional output space. The output space consists of a certain number of nodes (sometimes also called units or models), which are often arranged as a two-dimensional grid, in rectangular or hexagonal shape. One important property of the SOM is the fact that it preserves the topology of the input space as faithfully as possible, i.e. data that is similar and thus close to each other in the input space will also be located in vicinity in the output map. The SOM thus can be used to uncover complex inherent structures and correlations in the data, which makes it an attractive tool for data analysis.

The SOM has been applied in many Digital Library settings, to provide a novel, alternative way for browsing the library's content. This concept has also been applied for Music Retrieval to generate *music maps*, such as in the SOMEJB [2] system. Specific domain applications of music maps are for example the *Map of Mozart* [3], which organises the complete works of Mozart in an appealing manner, or the *Radio SOM* [4], illustrating musical profiles of radio stations. A comprehensive overview on music maps, with a special focus on the user interaction with them, can be found in [5].

### III. AUDIO FEATURES

In our experiments, we employ several different sets of features extracted from the audio content of the music, and compare them to each other. Specifically, we use the MARSYAS, Chroma, Rhythm Patterns, Statistical Spectrum Descriptors, and Rhythm Histograms audio feature sets, all of which will be described below.

#### A. MARSYAS Features

The MARSYAS system [6] is a software framework for audio analysis, feature extraction and retrieval. It provides a number of feature extractors that can be divided into three groups: features describing the timbral texture, those capturing the rhythmic content, and features related to pitch content.

The *STFT-Spectrum based Features* provide standard temporal and spectral low-level features, such as Spectral Centroid, Spectral Rolloff, Spectral Flux, Root Mean Square (RMS) energy and Zero Crossings. Further, MARSYAS computes the first twelve Mel-frequency cepstral coefficients (MFCCs).

The rhythm-related features aim at representing the regularity of the rhythm and the relative saliences and periods of diverse levels of the metrical hierarchy. They are based on the *Beat Histogram*, a particular rhythm periodicity function representing beat strength and rhythmic content of a piece of music. Various statistics are computed of the histogram: the relative amplitude of the first and second peak, the ratio of the amplitude of the second peak and the first peak, the period of the first and second beat (in beats per minute), and the overall sum of the histogram, as indication of beat strength.

The *Pitch Histogram* is computed by decomposing the signal into two frequency bands, for each of which amplitude envelopes are extracted and summed up, and the main pitches are detected. The three dominant peaks are accumulated into the histogram, containing information about the pitch range of a piece of music. A folded version of the histogram, obtained by mapping the notes of all octaves onto a single octave, contains information about the pitch classes or the harmonic content. The amplitude of the maximum peak of the folded histogram (i.e. magnitude of the most dominant pitch class), the period of the maximum peak of the unfolded (i.e. octave range of the dominant pitch) and folded histogram (i.e. main pitch class), the pitch interval between the two most prominent peaks of the folded histogram (i.e. main tonal interval relation) and the overall sum of the histogram are computed as features.

#### B. Chroma Features

Chroma features [7] aim to represent the harmonic content (e.g. keys, chords) of a short-time window of audio by computing the spectral energy present at frequencies that correspond to each of the 12 notes in a standard chromatic scale (e.g., black and white keys within one octave on a piano). We employ the feature extractor implemented in the MARSYAS system, and compute four statistical values for each of the 12-dimensional Chroma features, thus resulting in a 48-dimensional feature vector.

#### C. Rhythm Patterns

Rhythm Patterns (RP) are a feature set for handling audio data based on analysis of the spectral audio data and psycho-acoustic transformations [8], [9].

In a pre-processing stage, multiple channels are averaged to one, and the audio is split into segments of six seconds, possibly leaving out lead-in and fade-out segments.

The feature extraction process for a Rhythm Pattern is then composed of two stages. For each segment, the spectrogram of the audio is computed using the short time Fast Fourier Transform (STFT). The window size is set to 23 ms (1024 samples) and a Hanning window is applied using 50 % overlap between the windows. The Bark scale, a perceptual scale which groups frequencies to critical bands according to perceptive pitch regions [10], is applied to the spectrogram, aggregating it to 24 frequency bands. Then, the Bark scale spectrogram is transformed into the decibel scale, and further psycho-acoustic transformations are applied: computation of the Phon scale incorporates equal loudness curves, which account for the different perception of loudness at different frequencies [10]. Subsequently, the values are transformed into the unit Sone. The Sone scale relates to the Phon scale in the way that a doubling on the Sone scale sounds to the human ear like a doubling of the loudness. This results in a psycho-acoustically modified Sonogram representation that reflects human loudness sensation.

In the second step, a discrete Fourier transform is applied to this Sonogram, resulting in a (time-invariant) spectrum of loudness amplitude modulation per modulation frequency for each individual critical band. After additional weighting and smoothing steps, a Rhythm Pattern exhibits magnitude of modulation for 60 modulation frequencies (between 0.17 and 10 Hz) on 24 bands, and has thus 1440 dimensions.

In order to summarise the characteristics of an entire piece of music, the feature vectors derived from its segments are averaged by computing the median.

#### D. Statistical Spectrum Descriptors

Computing Statistical Spectrum Descriptors (SSD) features relies on the first stage of the algorithm for computing RP features. Statistical Spectrum Descriptors are based on the Bark-scale representation of the frequency spectrum. From this representation of perceived loudness, a number of statistical measures is computed per critical band, in order to describe fluctuations within the critical bands. Mean, median, variance, skewness, kurtosis, min- and max-value are computed for each of the 24 bands, and a Statistical Spectrum Descriptor is extracted for each selected segment. The SSD feature vector for a piece of audio is then calculated as the median of the descriptors of its segments.

In contrast to the Rhythm Patterns feature set, the dimensionality of the feature space is much lower – SSDs have  $24 \times 7 = 168$  instead of 1440 dimensions – at matching performance in terms of genre classification accuracies [9].

### E. Rhythm Histogram Features

The Rhythm Histogram features are a descriptor for the rhythmic characteristics in a piece of audio. Contrary to the Rhythm Patterns and the Statistical Spectrum Descriptor, information is not stored per critical band. Rather, the magnitudes of each modulation frequency bin (at the end of the second phase of the RP calculation process) of all 24 critical bands are summed up, to form a histogram of ‘rhythmic energy’ per modulation frequency. The histogram contains 60 bins which reflect modulation frequency between 0.168 and 10 Hz. For a given piece of audio, the Rhythm Histogram feature set is calculated by taking the median of the histograms of every 6 second segment processed.

### IV. COMPARISON OF SELF-ORGANISING MAPS

Self-Organising Maps can differ from each other depending on a range of various factors: simple ones such as different initialisations of the random number generator, to more SOM-specific ones such as different parameters for e.g. the learning rate and neighbourhood kernel (cf. [1] for details), to differences in the map-size. In all such cases, the general topological ordering of the map should stay approximately the same, i.e. clusters of data items would stay in the neighbourhood of similar clusters, and be further away from dissimilar ones, unless the parameters were chosen really bad. Still, some differences will appear, which might then range from e.g. a minor deviation such as a mirrored arrangement of the vectors on the map, to having still the same local neighbourhood between specific clusters, but a slightly rotated or skewed global layout. Training several maps with different parameters and then analysing the differences can thus give vital clues on the structures inherent in the data, by discovering which portions of the input data are clustered together in a rather stable fashion, and for which parts random elements play a vital role for the mapping.

An analytic method to compare different Self-Organising Maps, created with such different training parameters, but also with different sizes of the output space, or even with different feature sets, has been proposed in [11]. For the study presented in this paper, especially the latter, comparing different feature sets, is of major interest. The method allows to compare a selected *source* map to one or more *target* maps by comparing how the input data items are arranged on the maps. To this end, it is determined whether data items located close to each other in the source map are also closely located to each other in the target map(s), to determine whether there are *stable* or *outlier* movements between the maps. “Close” is a user-adjustable parameter, and can be defined to be on the same node, or within a certain radius around the node. Using different radii for different maps accommodates for maps differing in size. Further, a higher radius allows to see a more abstract, coarse view on the data movement. If the majority of the data items stays within the defined radius, then this is regarded a stable shift, or an outlier shift otherwise. Again, the user can specify how big the percentage needs to be to regard it a stable or outlier shift. These shifts are visualised by arrows, where

different colours indicate stable or outlier shifts, and the line-width determines the cardinality of the data items moving along the shift. The visualisation is thus termed *Data Shifts* visualisations. Figure 1 illustrates stable (green arrows) and outlier (red arrows) shifts on selected nodes of two maps, the left one trained with Rhythm Pattern, the right one with SSD features. Already from this illustration, we can see that some data items will also be closely located on the SSD map, while others spread out to different areas of the map.

Finally, all these analysis steps can be done as well not on a per-node basis, but rather regarding clusters of nodes instead. To this end, first a clustering algorithm is applied to the two maps to be compared to each other, to compute the same, user-adjustable number of clusters. Specifically, we use Ward’s linkage clustering [12], which provides a hierarchy of clusters at different levels. The best-matching clusters found in both SOMs are then linked to each other, determined by the highest matching number of data points for pairs of clusters on both maps – the more data vectors from cluster  $A_i$  in the first SOM are mapped into cluster  $B_j$  in the second SOM, the higher the confidence that the two clusters correspond to each other. Then all pairwise confidence values between all clusters in the maps are computed. Finally, all pairs are sorted and repeatedly the match with the highest values is selected, until all clusters have been assigned exactly once. When the matching is determined, the *Cluster Shifts* visualisation can easily be created, analogous to the visualisation of Data Shifts.

An even more aggregate and abstract view on the input data movement can be provided by the *Comparison Visualisation*, which further allows to compare one SOM to several other maps in the same illustration. To this end, the visualisation colours each unit  $u$  in the main SOM according to the average pairwise distance between the unit’s mapped data vectors in the other  $s$  SOMs. The visualisation is generated by first finding all  $k$  possible pairs of the data vectors on  $u$ , and compute the distances  $d_{ij}$  of the pair’s positions in the other SOMs. These distances are then summed and averaged over the number of pairs and the number of compared SOMs, respectively. Alternatively to the mean, the variance of the distances can be used.

### V. ANALYTIC COMPARISON OF AUDIO FEATURE SETS

In this section, we outline the results of our study on comparing the different audio feature sets with Self-Organising Maps.

#### A. Test Collection

We extracted features for the collection used in the ISMIR 2004 genre contest<sup>1</sup>, which we further refer to as *ISMIRgenre*. The dataset has been used as benchmark for several different MIR systems. It comprises 1458 tracks, organised into six different genres. The greatest part of the tracks belongs to Classical music (640, colour-coded in red), followed by World (244, cyan), Rock/Pop (203, magenta), Electronic (229, blue), Metal Punk (90, yellow), and finally Jazz/Blues (52, green).

<sup>1</sup>[http://ismir2004.ismir.net/ISMIR\\_Contest.html](http://ismir2004.ismir.net/ISMIR_Contest.html)

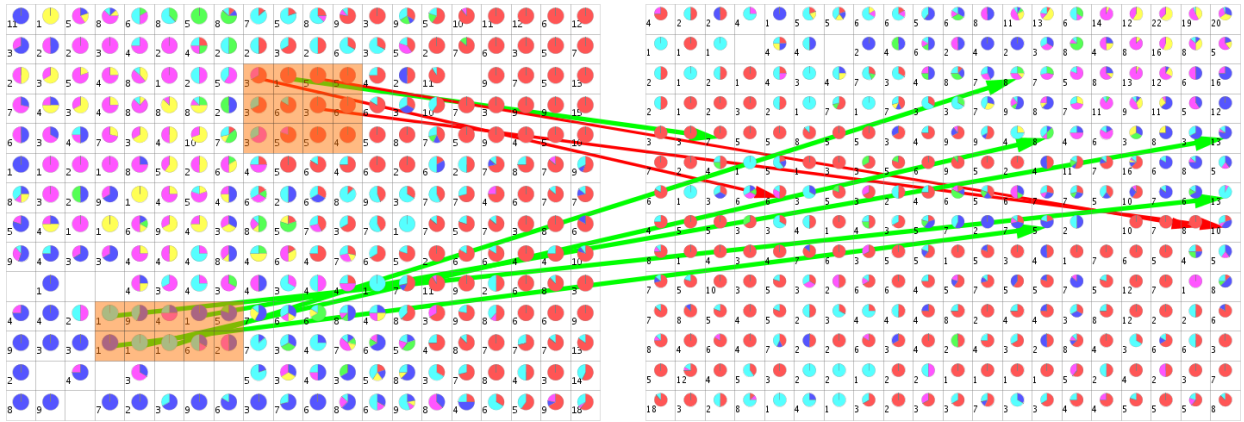


Fig. 1. Data Shifts Visualisation for RP and SSD maps on the ISMIRgenre data sets

TABLE I  
CLASSIFICATION ACCURACIES ON THE ISMIRGENRE DATABASES.

Feature Set	1-nn	3-nn	Naïve B.	SVM
Chroma	39.59	45.54	40.73	<i>45.07</i>
Rhythm Histograms	60.45	63.04	56.74	<i>63.74</i>
MARSYAS	66.69	64.36	59.00	<i>67.02</i>
Rhythm Patterns	73.21	71.37	63.31	<i>75.17</i>
SSD	<b>78.20</b>	<b>76.44</b>	<b>60.88</b>	<i>78.73</i>

### B. Genre Classification Results

To give a brief overview on the discriminative power of the audio feature sets, we performed a genre classification on the collection, using the WEKA machine learning toolkit<sup>2</sup>. We utilised k-Nearest-Neighbour, Naïve Bayes and Support Vector Machines, and performed the experiments based on a ten-fold cross-validation, which is further averaged over ten repeated runs. The results given in Table I are the micro-averaged classification accuracies.

There is a coherent trend across all classifiers. It can be noted that SSD features are performing best on each single classifier (indicated by bold print), achieving the highest value with Support Vector Machines, followed surprisingly quite closely by 1-nearest-neighbours. Also the subsequent ranks don't differ across the various classifiers, with Rhythm Patterns being the second-best feature sets, followed by MARSYAS, Rhythm Histograms and the Chroma features. In all cases, SVM are the dominant classifier (indicated by italic type), with the k-NN performing not that far off of them. These results are in line with those previously published in the literature.

### C. Genre Clustering with Music Maps

We trained a number of Self-Organising Maps, with different parameters for the random number generator, the number of training iterations, and in different size, for each of the five feature sets. An interesting observation is the arrangement of the different genres across the maps, which is illustrated in Figure 2. While the different genres form pretty clear and

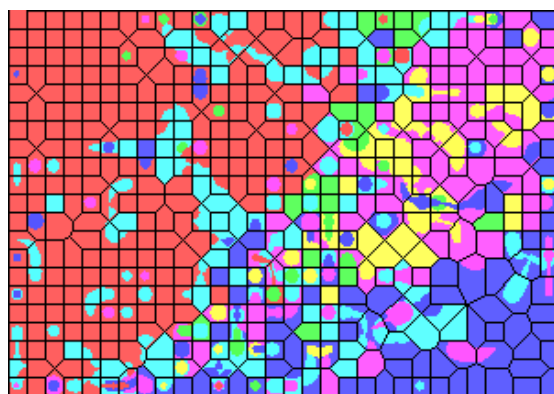
distinct clusters on RP, RH and SSD features this is not so much the case for Chroma or MARSYAS features. Figure 2(a) shows the map on RP features. It can be quickly observed that the genres Classical (red), Electronic (blue) and Rock/Pop (magenta) are clearly arranged closely to each other on the map; also Metal/Punk (yellow) and Jazz/Blues (green) are arranged on specific areas of the map. Only World Music (cyan) is spread over many different areas; however, World Music is rather a collective term for many different types of music, thus this behaviour seems not surprising. The maps for RH and SSD features exhibit a very similar arrangement.

For the MARSYAS maps, a pre-processing step of normalising the single attributes was needed, as otherwise, different value ranges of the single features would have a distorting impact on distance measurements, which are an integral part of the SOM training algorithm. We tested both a *standard score* normalisation (i.e. subtracting the mean and dividing by the standard deviation) and a min-max normalisation (i.e. values of range [0..1] for each attribute). Both normalisation methods dramatically improved the subjective quality of the map, both showing similar results. Still, the map trained with the MARSYAS features, depicted in Figure 2(b), shows a less clear clustering according to the pre-defined genres. The Classical genre occupies a much larger area, and is much more intermingled with other genres, and actually divided in two parts by genres such as Rock/Pop and Metal/Punk. Also, the Electronic and Rock/Pop genres are spread much more over the map than with the RP/RH/SSD features. A subjective evaluation by listening to some samples of the map also found the RP map to be superior in grouping similar music.

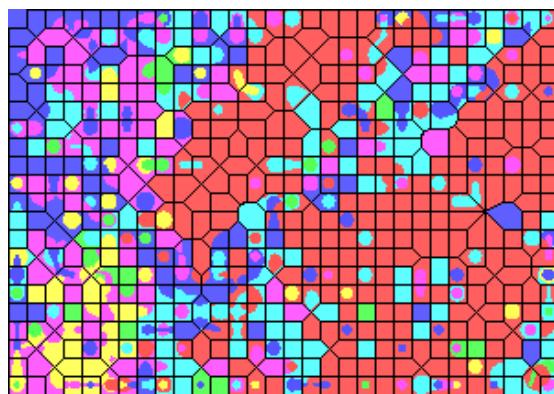
Similar observations hold also true for all variations of parameters and sizes trained, and can further be observed for maps trained on Chroma features.

Thus, a first surprising finding is that MARSYAS features, even though they provide good classification results, outperforming RH features on all tested classifiers and not being that far off from the results with RP, are not exhibiting properties that would allow the SOM algorithm to cluster them as well as with the other feature sets.

<sup>2</sup><http://www.cs.waikato.ac.nz/ml/weka/>



(a) Rhythm Patterns



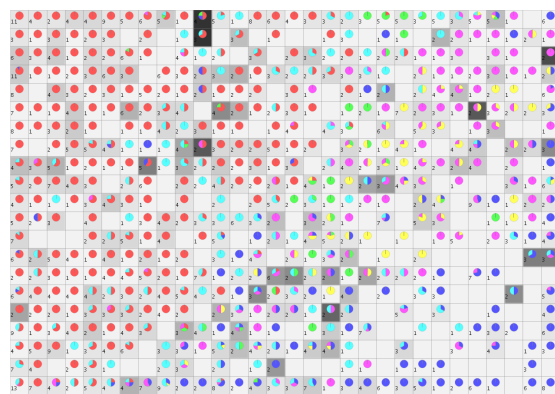
(b) MARSYAS

Fig. 2. Distribution of genres on two maps trained with the same parameters, but different feature sets

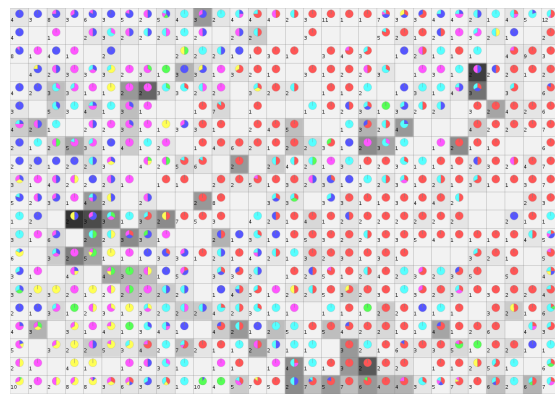
#### D. Mapping Stability

Next, we present the analysis of the stability of the mapping on single feature sets, i.e., we compare maps trained with the same feature sets, but different parameters, to each other. One such visualisation is depicted in Figure 3(a), which compares maps trained with RP features. The darker the nodes on the map, the more instable the mapping of the vectors assigned to these nodes is in regard to the other maps compared to. We can see that quite a big area of the map seems to be pretty stable in mapping behaviour, and there are just a few areas that get frequently shuffled on the map. Most of those are in areas that are the borderlines between clusters that each contain music from a specific genre. Among those, an area in the upper-middle border of the map holds musical pieces from Classical, Jazz/Blues, Electronic, and World Music genres. Two areas, towards the right-upper corner, are at intersections of Metal/Punk and Pop/Rock genres, and frequently get mapped into slightly different areas on the map. We further trained a set of smaller maps, on which we observed similar patterns.

While the SOMs trained with the MARSYAS features are not preserving genres topologically on the map, the mapping itself seems to be stable, as can be seen in Figure 3(b). From a visual inspection, it seems there are not more “instable” areas on the map than with the RP features, and as well, they can be mostly found in areas where genre-clusters intermingle.



(a) Rhythm Patterns



(b) MARSYAS

Fig. 3. Comparison of the two maps from Figure 2 to other maps trained on the same respective feature set, but with different training parameters

#### E. Feature Comparison

Finally, we want to compare maps trained on different feature sets. Figure 4 shows a comparison of an RP with an SSD map, both of identical size. The Rhythm Patterns map is expected to cover both rhythm and frequency information from the music, while the Statistical Spectrum Descriptors are only containing information on the power spectrum. Thus, an increased number of differences in the mapping is expected when comparing these two maps, in contrast to a comparison of maps trained with the same feature set. This hypothesis is confirmed by a visual inspection of the visualisation, which shows an increased amount of nodes colour-coded to have high mapping distances in the other map.

Those nodes are the starting point for investigating how the pieces of music get arranged on the maps. In Figure 4, a total of four nodes, containing two tracks each, have been selected in the left map, trained with the RP features. In the right map, trained with the SSD features, the grouping of the tracks is different, and no two tracks got matched on the same node or even neighbourhood there. Rather, from both the lower-leftmost and upper-rightmost node containing Classical music, one track each has been grouped together closely at the centre-right area and at the left-centre border. Likewise, the other two selected nodes, one containing World Music, the other World Music and Classical Music, split up in a similar fashion. One

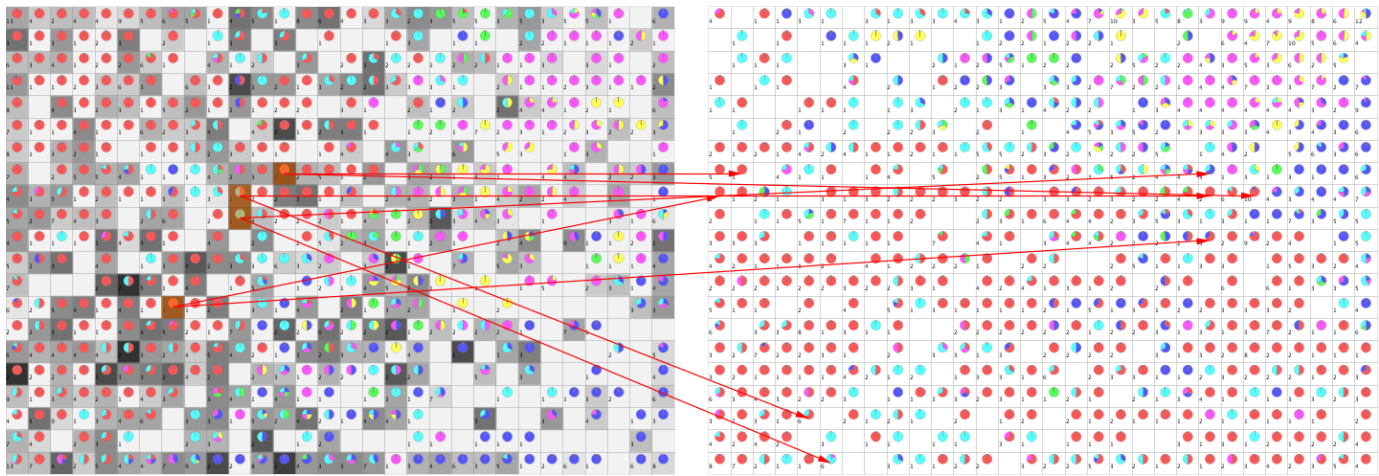


Fig. 4. Comparison of a RP and an SSD map

track each gets mapped to the lower-left corner, at the border of the Classical and World Music cluster. The other two tracks lie in the centre-right area, close to the other two tracks mentioned previously.

Manually inspecting the new clustering of the tracks on the SSD based maps reveals that in all cases, the instrumentation is similar within all music tracks. However, on the RP map, the music is as well arranged by the rhythmic information captured by the feature set. Thus the tracks located on the same node in the left map also share similar rhythmic characteristics, while this is not necessarily the case for the right, SSD-based map.

To illustrate this in more detail, one of the Classical music pieces in the lower-left selected node on the RP map is in that map located clearly separated from another Classical piece on the upper one of the two neighbouring selected nodes in the centre. Both tracks exhibit the same instrumentation, a dominant violin. However, the two songs differ quite strongly in their tempo and beat, the latter music piece being much more lively, while the first has a much slower metre. This differentiates them in the Rhythm Pattern map. However, in the Statistical Spectrum Descriptors, rhythmic characteristics are not considered in the feature set, and thus these two songs are correctly placed in close vicinity of each other.

Similar conclusions can be drawn for comparing other feature sets. An especially interesting comparison is Rhythm Patterns vs. a combination of SSD and Rhythm Histogram features, which together cover very similar characteristics as the Rhythm Patterns, but still differ e.g. in classification results. Also, comparing Rhythm Patterns or Histograms to MARSYAS offers interesting insights, as they partly cover the same information about music, but also have different features.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, we utilised Self-Organising Maps to compare five different audio feature sets regarding their clustering characteristics. One interesting finding was that maps trained with MARSYAS features are not preserving the pre-defined ordering into genres as well as it is the case with RP, RH and

SSD features, even though they are similar in classification performance. Further, we illustrated that the approach of using Self-Organising Maps for an analytical comparison of the feature sets can provide vital clues on which characteristics are captured by the various audio feature sets, by highlighting which pieces of music are interesting for a closer inspection. One challenge for future work is to automate detecting interesting patterns as presented in the previous Section.

## REFERENCES

- [1] T. Kohonen, *Self-Organizing Maps*, ser. Springer Series in Information Sciences. Berlin, Heidelberg: Springer, 1995, vol. 30.
- [2] A. Rauber, E. Pampalk, and D. Merkl, "The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models," *Journal of New Music Research*, vol. 32, no. 2, June 2003.
- [3] R. Mayer, T. Lidy, and A. Rauber, "The map of Mozart," in *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR'06)*, October 8-12 2006.
- [4] T. Lidy and A. Rauber, "Visually profiling radio stations," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, October 8-12 2006.
- [5] J. Frank, T. Lidy, E. Peiszer, R. Genswaidner, and A. Rauber, "Creating ambient music spaces in real and virtual worlds," *Multimedia Tools and Applications*, vol. 44, no. 3, 2009.
- [6] G. Tzanetakis and P. Cook, "Marsyas: A framework for audio analysis," *Organized Sound*, vol. 4, no. 30, 2000.
- [7] M. Goto, "A chorus section detection method for musical audio signals and its application to a music listening station," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 14, no. 5, 2006.
- [8] A. Rauber, E. Pampalk, and D. Merkl, "Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by musical styles," in *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'02)*, Paris, France, October 13-17 2002.
- [9] T. Lidy and A. Rauber, "Evaluation of feature extractors and psycho-acoustic transformations for music genre classification," in *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05)*, London, UK, September 11-15 2005.
- [10] E. Zwicker and H. Fastl, *Psychoacoustics, Facts and Models*, 2nd ed., ser. Series of Information Sciences. Berlin: Springer, 1999, vol. 22.
- [11] R. Mayer, D. Baum, R. Neumayer, and A. Rauber, "Analytic comparison of self-organizing maps," in *Proceedings of the 7th Workshop on Self-Organizing Maps (WSOM'09)*, St. Augustine, FL, USA, June 8-10 2009.
- [12] J. H. Ward, Jr., "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, vol. 58, no. 301, March 1963.