Musical style classification from symbolic data: A two-styles case study

Pedro J. Ponce de León and José M. Iñesta

Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante,
Ap. 99, E-03080 Alicante, Spain
{pierre,inesta}@dlsi.ua.es

Abstract. In this paper the classification of monophonic melodies from two different musical styles (Jazz and classical) is studied using different classification methods: Bayesian classifier, a k-NN classifier, and self-organising maps (SOM). From MIDI files, the monophonic melody track is extracted and cut into fragments of equal length. From these sequences, A number of melodic, harmonic, and rhythmic numerical descriptors are computed and analysed in terms of separability in two music classes, obtaining several reduced descriptor sets. Finally, the classification results for each type of classifier for the different descriptor models are compared. This scheme has a number of applications like indexing and selecting musical databases or the evaluation of style-specific automatic composition systems.
Keywords: music information retrieval, self-organising maps, bayesian classifier, nearest neighbours (k-NN), feature selection.

1 Introduction

The automatic machine learning and pattern recognition techniques, successfully employed in other fields, can be also applied in music analysis. One of the tasks that can be posed is the modelization of the music style. Immediate applications are the classification, indexation and content-based search in digital music libraries, where digitised (MP3), sequenced (MIDI) or structurally represented (XML) music can be found. The computer could be trained in the user musical taste in order to look for that kind of music over large musical databases. Such a model could also be used in cooperation with automatic composition algorithms to guide this process according to a stylistic profile provided by the user.

Our aim is to develop a system able to distinguish musical styles from a symbolic representation of a melody using musicological features: melodic, harmonic and rhytmic ones. Our working hypothesis is that melodies from a same musical genre may share some common features that permits to assign a musical style to them. For testing our approach, we have initially chosen two music styles, jazz and classical, for our experiments. We will also investigate whether such a representation by itself has enough information to achieve this goal or, on the contrary, also timbric information has to be included for that purpose.

In this work we will start with some related works in the area of musical style identification. Then our methodology will be presented, describing the musical data and the statistical description model we have used. Next, the implementation and parametrization for the different classification methods we used, namely Bayesian classifier, k-nearest neigbours (kNN) and self-organising maps (SOM) will be briefly explained. The initial set of descriptors will be statistically analized to test their contribution to the musical style separability. These procedures will lead us to reduced models, discarding not useful descriptors. Then, the SOM training and the classification results obtained with each classifier will be presented. These results are compared and the advantages and drawbacks of these classification methods related to the musical classification task are discussed. Finally, conclusions and current and future lines of work are presented.

1.1 Related work

A number of recent papers explore the capabilities of SOM to analyse and classify music data. Rauber and Frühwirth [1] pose the problem of organising music digital libraries according to sound features of musical themes, in such a way that similar themes are clustered, performing a content-based classification of the sounds. Whitman and Flake [2] present a system based on neural nets and support vector machines, able to classify an audio fragment into a given list of sources or artists. Also in [3], the authors describe a neural system to recognise music types from sound inputs. In [4] the authors present a hierarchical SOM able to analyse time series of musical events and then discriminate those events in a different musical context. In the work by Thom [5] pitch histograms (measured in semitones relative to the tonal pitch and independent of the octave) are used to describe blues fragments of the saxophonist Charlie Parker. The pitch frequencies are used to train a SOM. Also pitch histograms and SOM are used in [6] for musicological analysis of folk songs.

These works pose the problem of music analysis and recognition using either digital sound files or symbolic representations as input. The approach we propose here is to use the symbolic representation of music that will be analysed to provide melodic, harmonic and rhythmic descriptors as input to the Bayesian, kNN and SOM classifiers (see Fig. 1) for classification of musical fragments into one of two musical styles. We use standard MIDI files as the source of monophonic melodies.

2 Methodology

2.1 Musical data

MIDI files from two styles, jazz and classical music, were collected. Classical music was chosen and melodic samples were taken from works by Mozart, Bach, Schubert, Chopin, Grieg, Vivaldi, Schumann, Brahms, Beethoven, Dvorak, Haendel, Paganini and Mendhelson. Jazz music consist of jazz standard tunes from

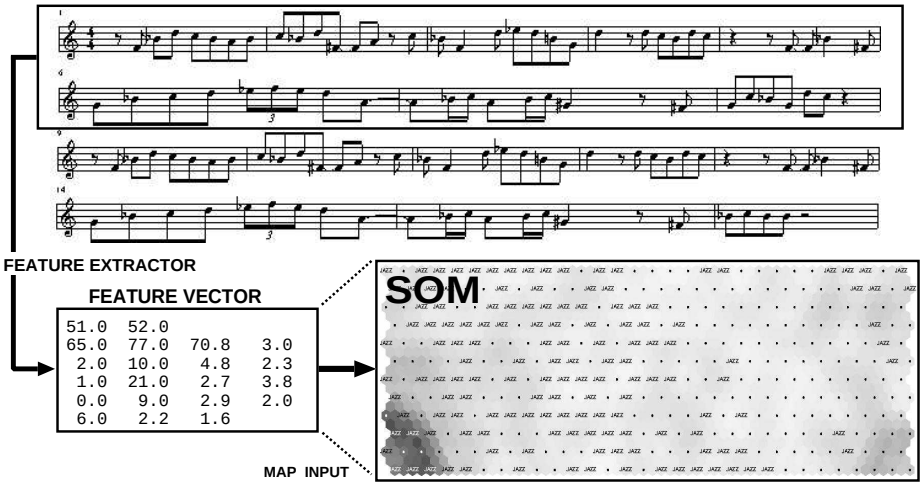

Fig. 1. Structure of the system: musical descriptors are computed from a window 8-bar wide and provided to a classifier (a SOM in this figure) for training and classification. Once trained, a style label is assigned to the units. During classification, the label of the winning unit provides the style to which the music fragment belongs to. This example is based on the Charlie Parker's jazz piece "Dexterity".

a variety of authors like Charlie Parker, Duke Ellington, Bill Evans, Miles Davis, etc. The monophonic melodies are isolated from the rest of the musical content in these MIDI files —the track number of the melody been known for each file—. This way, we get a sequence of musical events that can be either notes or silences. Other kind of MIDI events are filtered out.

Here we will deal only with melodies written in 4/4. In order to have more restricted data, each melody sequence is divided into fragments of 8 bars. We assume this length suffices to get a good sense of the melodic phrase in the context of a 4/4 signature. Each of these fragments becomes a data sample. No segmentation techniques were applied to obtain the samples. You can compare this straightforward process as tuning a radio station, hear the music for a while and then switching off your radio box. Most people would be able to identify the music style they heard (provided they are familiar with that music style).

Each melodic sample is represented by a number of statistical descriptors (see section 2.2). The total number of samples is 1244, and approximately a 40% are jazz samples and a 60% are classical samples.

2.2 Feature extraction

We have chosen a vector of musical descriptors of the melodic fragments as the input for the classifiers, instead of the explicit representation of the melodies. Thus, a description model is needed. Firstly, three groups of features are extracted: melodic, harmonic and rhythmic properties.

The melody tracks in the MIDI files are quantised in origin at 480 pulses per bar (i.e., real-time quantisation), therefore possibly containing some amount of phrasing like swing in jazz tunes, or stacatto/legato parts in classical music. The features are computed using a time resolution of $Q = 48$ pulses per bar, down-quantising the original melody track. This resolution is the minimum common multiple of the most common divisors of the whole note: 2,3,4,6,8,12,16, and permits capturing the most of the rythmic content in binary and ternary form (i.e., triplets). We consider this enough for the purpose of music style classification. A higher resolution would capture shorter events, particularly very short silences between much larger notes that are not real musical silences, but rather short gaps between notes due to the particular musical technique of the interpreter. On the other hand, a lower resolution would miss 8th triplets or 16th note durations, that are undoubtely important to be considered for style characterization.

The initial set of 22 musical descriptors is:

- Overall descriptors:
  - Number of notes and number of silences in the fragment.
- Pitch descriptors:
  - Lowest, highest (provide information about the pitch range of the melody), average, and standard deviation (provide information about how the notes are distributed in the score).
- Note duration descriptors (these descriptors are measured in pulses):
  - Minimum, maximum, average, and standard deviation.
- Silence duration descriptors (in pulses):
  - Minimum, maximum, average, and standard deviation.
- Interval descriptors (distance in pitch between two consecutive notes):
  - Minimum, maximum, average, and standard deviation.
- Harmonic descriptors:
  - *Number of non diatonic notes.* An indication of frequent excursions outside the song key (extracted from the MIDI file) or modulations.
  - *Average degree of non diatonic notes.* Describes the kind of excursions. Its a number between 0 and 4 that indexes the non diatonic notes of the diatonic scale of the tune key, that can be major or minor key[1]. It can take a fractional value.
  - *Standard deviation of degrees of non diatonic notes.* Indicates a higher variety in the non diatonic notes.
- Rhytmic descriptor: *number of syncopations*: notes not beginning at the rhythm beats but in some places between them (usually in the middle) and that extend across beats. This is actually an estimation of the number of syncopations. We are not interested in the exact number of them for our task.

In section 2.4 a feature selection procedure is presented in order to obtain some reduced description models and to test their classification ability.

[1] 0: ♭II, 1: ♭III (♮III for minor key), 2: ♭V, 3: ♭VI, 4: ♭VII.

We used the key meta-event present in each MIDI file to compute the harmonic descriptors. The correctness of its value was verified for each file prior to the feature extraction process.

With this set of descriptors, we assume the following hypothesis: melodies of the same style are closer to each other in the description space than melodies from different styles. We will test the performance of different classifiers to verify this hypothesis.

This kind of statistical description of musical content is sometimes referred to as *shallow structure description* [7]. It is similar to histogram-based descriptions, like the one found in [6], that it tries to model the distribution of musical events in a musical fragment. Computing the minimum, maximum, mean and standard deviation from the distribution of musical features like pitches, durations, intervals and non-diatonic notes we reduce the number of features needed (each histogram may be made up of tens of features), while assuming the distributions for the mentioned features to be normal within a melodic fragment. Other authors have also used some of the descriptors presented here to classify music [8].

2.3 Classifier implementation and tunning

Three different classifiers are used in this paper to automatic style identification. Two of them are supervised methods: The bayesian classifier and the kNN classifier [9]. The other one is an unsupervised learning neural network, the self-organising map (SOM) [10]. SOM are neural methods able to obtain approximate projections of high-dimensional data distributions into low-dimensional spaces, usually bidimensional. With the map, different clusters in the input data can be located. These clusters can be semantically labelled to characterise the training data and also hopefully future new inputs.

For the Bayesian classifier, we assume that individual descriptor probability distributions for each style are normal, with means and variances estimated from the training data. This classifier computes the squared Mahalanobis distance from test samples to the mean vector of each style in order to obtain a classification criterion. The kNN classifier uses an euclidean metrics to compute distance between samples, and we tested a number of odd values for k, ranging from 1 to 25.

For SOM implementation and graphic representations the SOM_PAK software [11] has been used. For the experiments, two different geometries were tested: 16×8 and 30×12 maps. An hexagonal topology for unit connections and a bubble neighbourhood for training have been selected. The value for this neighbourhood is equal for all the units in it and decreases as a function of time. The training was done in two phases. See Table 1 for the different training parameters. The metrics used to compute distance between samples was again the euclidean distance.

In the next pages, the maps are presented using the U-map representation, where the units are displayed by hexagons with a dot or label in their centre. The grey level of unlabelled hexagons represents the distance between neighbour

Table 1. SOM training parameters

Map size	16×8	30×12
First training phase (coarse ordering)		
Iterations	3,000	10,000
Learning rate	0.1	0.1
Neigbourhood radius	12	20
Second training phase (fine tunning)		
Iterations	30,000	100,000
Learning rate	0.05	0.05
Neigbourhood radius	4	6

units (the clearer the closer they are). For the labelled units is an average of the neighbour distances. This way, clear zones are clusters of units, sharing similar weight vectors. The labels are a result of callibrating the map with a series of test samples and indicate the class of samples that activates more times each unit.

2.4 Feature selection procedure

We have devised a selection procedure in order to keep those descriptors that actually contribute to make the classification. The procedure is based on the values for all the features in the weight vectors of five previously trained SOMs. The maps are trained and labelled (calibrated) in an unsupervised manner (see Fig 2-a for an example). We try to find which descriptors provide more useful information for the classification. Some descriptor values for the weight vectors correlate better than others with the label distribution in the map. It is reasonable to consider that these descriptors contribute more to achieve a good separation between classes. See Fig. 2-b and 2-c for descriptor planes that correlate and that do not with the class labels.

Consider that the N descriptors are random variables $\{x_i\}_{i=1}^{N}$ that corresponds to the weight vector components for each of the M units in the map. We drop the subindex i for clarity, because all the discussion is related to each descriptor. We will divide the set of M values for each descriptor into two subsets: $\{x_j^C\}_{j=1}^{M_C}$ are the descriptor values for the units labelled with the classical style and $\{x_j^J\}_{j=1}^{M_J}$ are those for the jazz units, being M_C and M_J the number of units labelled with classical and jazz labels, respectively. We want to know whether these two set of values follow the same distribution or not. If false, it is an indication that there is a clear separation between the values of this descriptor for the two classes, so it is a good feature for classification and should be kept in the model and otherwise it does not seem to provide separability to the classes.

We assumed that both sets of values hold normality conditions and the following statistical for sample separation has been applied:

$$z = \frac{|\bar{x}_C - \bar{x}_J|}{\sqrt{\frac{s_C^2}{M_C} + \frac{s_J^2}{M_J}}} \quad , \tag{1}$$

where $\bar{x}_C$ and $\bar{x}_J$ are the means, and s_C^2 and s_J^2 the variances for the descriptor values for both classes. The larger the z value is, the higher the separation between both sets of values is for that descriptor. This value permits to order the descriptors according to their separation ability and a threshold can be established to determine which descriptors are suitable for the model. This threshold, computed from a t-student distribution with infinite degrees of freedom and a 99.5% confidence interval, is $z = 2.81$.

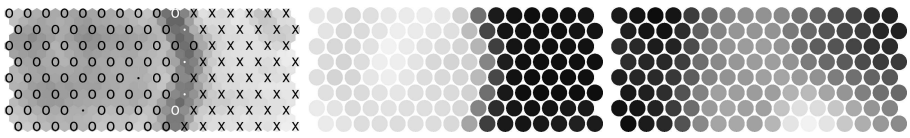

Fig. 2. Contribution to classification: (a:left) callibrated map ('X' and 'O' are the labels for both styles); (b:center) weight space plane for a feature that correlates with the areas; (c:right) plane for a feature that does not correlate.

3 Experiments and results

As stated above, we have chosen two given music styles: jazz and classical for testing our approach. The jazz samples were taken from jazz standards from different jazz styles like be-bop, hard-bop, big-band swing, etc., and the melodies were sequenced in real time. Classical tunes were collected from a number of styles like baroque, romantic, renaissance, impressionism, etc.

The first step was to train SOMs with the whole set of descriptors. After training and labelling, maps like that in figure 3 have been obtained. It is observed how the labelling process has located the jazz labels mainly on the left zone, and those corresponding to classical melodies on the right. Some units can be labelled for both music styles if they are activated by fragments from both styles. In these cases there is always a winner label (the one displayed) according to the number of activations. The proportion of units with both labels is the overlapping degree, that for the presented map was very low (8.0 %), allowing a clear distinction between styles.

In the Sammon projection of the map in figure 3 a knot separates two zones in the map. The zone at the left of the knot has a majority presence of units labelled with the jazz label and the zone at the right is mainly classical.

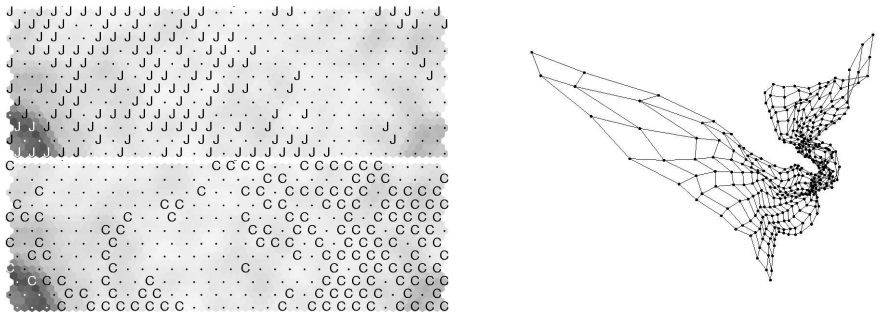

Fig. 3. Left: SOM map after being labeled with jazz (top) and classical (down) melodies. Note how both classes are clearly separated. Right: Sammon projection of the SOM, a way to display in 3D the organisation of the weight vector space.

3.1 Feature selection results

Firstly we have trained the maps with the whole set of 22 features. This way a reference performance for the system is obtained. In addition, we have trained other maps using just melodic descriptors and also melodic and harmonic ones. We get a set of five trained maps in order to study the values of the weight space planes, using the method described in 2.4. This number of experiments has been considered enough due to the repetitivity of the obtained results. For each experiment we have ordered the descriptors according to their value for z_i (see eq. 1). In table 2 the feature selection results are displayed, showing what descriptors have been considered for each model according to those results. Each model number denotes the number of descriptors included in that model. We have chosen four reduced model sizes: 6, 7, 10 and 13 descriptors. The 7-descriptor model includes the best rated descriptors. The 6-descriptor model excludes syncopation. The other two models include other worse rated descriptors.

Table 2. Feature selection results. For each model the descriptors included are shown in the right column.

Model	Descriptors
6	Highest pitch, max. interval, dev. note duration, max. note duration, dev. pitch, avg. note duration
7	+syncopation
10	+avg. pitch, dev. interval, number of notes
13	+number of silences, min. interval, num. non-diatonic notes
22	All the descriptors computed

3.2 Classification

For obtaining reliable results a scheme based on *leave-k-out* has been carried out. In our case $k = 10\%$ of the size of the whole database. This way, 10 sub-

experiments were performed for each experiment and the results have been averaged. In each experiment the training set was made of a different 90% of the total database and the other 10% was kept for testing.

The results obtained with the Bayesian classifier are presented in table 3.2. The best success rate was obtained with the 13-descriptor model (85.7%). Results from kNN classification can be seen in figure 4, with the error rate in function of the k parameter for each descriptor model. The minimum error rate was 0.113 (88.7% success) with $k = 9$ and a 7-descriptor model.

Table 3. Bayesian average success percentages.

Model	Jazz	Classic	Total
6	74.7	83.9	80.5
7	83.9	85.5	84.9
10	90.8	80.7	84.7
13	91.4	81.9	85.7
22	90.8	72.7	79.9

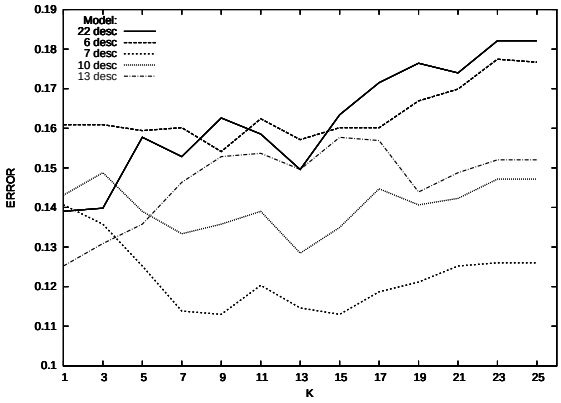

Fig. 4. kNN error rate with different k for each model.

Finally the results for the SOM classifier are presented in Table 4. These are average successful classification rate for Jazz and Classical samples. Each model has been evaluated with the two different SOM size SOM, and the best results presented here were consistently obtained with the 16×8 map geometry.

The best average results were obtained for that map when using the 7-descriptor model (84.2 %). It is observed that 6-descriptor model performance are systematically improved when syncopation is included in the 7-descriptor model. In some experiments even a 98.0 % of success (96.0 % for both styles) has been achieved. The inclusion of descriptors discarded in the feature selection

Table 4. Average success rate percentages for the SOM classifier

Model	Jazz	Classic	Total
6	79.4	82.1	80.8
7	81.8	86.5	84.2
10	78.8	82.7	80.7
13	72.0	82.6	77.3
22	72.7	79.6	76.1

test worsens the results and the worst case is when all of them are used (76.1 %). This is also true for the other classifiers presented here.

3.3 Result comparison

The three classifiers give comparable results. The better average results were achieved with the kNN classifier with $k = 9$ and the 7-descriptors model (88.7% of average success). The kNN classifier provided an almost perfect classification in some of the leaving-10%-out partitions of the 7-descriptor model data, with $k = 7$ (99.1% of success, that means that only one melody out 123 was misclassified). The bayesian classifier performed similarly with some data partitions for 10 and 13-descriptors models reaching 98.4% of success (only two misclassifications), having ranked second in the average success rate (85.7%). The best results for SOM were achieved with the 7-descriptors model, with up to a 95.9% success (five misclassifications), having the worst average success rate (84.2%, 7-descriptors model) of the three classifiers.

All these results are comparable, each of the classifiers presenting advantages and drawbacks. The Bayesian classifier is the fastest one, being theoretically the optimum classifier, if the data distributions are normal. One secondary result we obtained is the reinforcement of the normality assumption we made to select descriptors, since normality was a precondition to the Bayesian classifier we implemented.

The kNN classifier is conceptually simple and easy to implement, but computationally intensive (in time and space). It gave the best classification results, while it remains to be a supervised technique, like the Bayesian classifier.

Supervised learning can be a problem when you need an high amount of training data to be reasonably sure that you are covering the most of the input space. This is our situation, where the input space, monophonic (or even polyphonic) melodies of arbitrary length from many different musical styles distribute in a very large input space. So we have focused our attention on the SOM, that provides results slightly worse than the other classifiers, but it has the advantage of being an unsupervised classification method (you need only a very few labelled samples to calibrate the map). Furthermore, SOMs provide additional information thanks to their data visualization capabilities.

As an example of the visualization capabilities of the SOM, Figures 5 and 6 show how an entire melody is located in the map. One melody of each style is shown, the first 64 bars of the Allegro movement from the *Divertimento in D* by Mozart for the classical style, and three choruses of the standard jazz tune *Yesterdays*. The grey area in each map corresponds to the style the melody belongs to. The first map picture for each melody shows the map unit activated by the first fragment of the melody. The next pictures show the map unit activated by the next fragment of that melody. Consecutively activated units are linked by a straight line, displaying the path followed by the melody in the map.

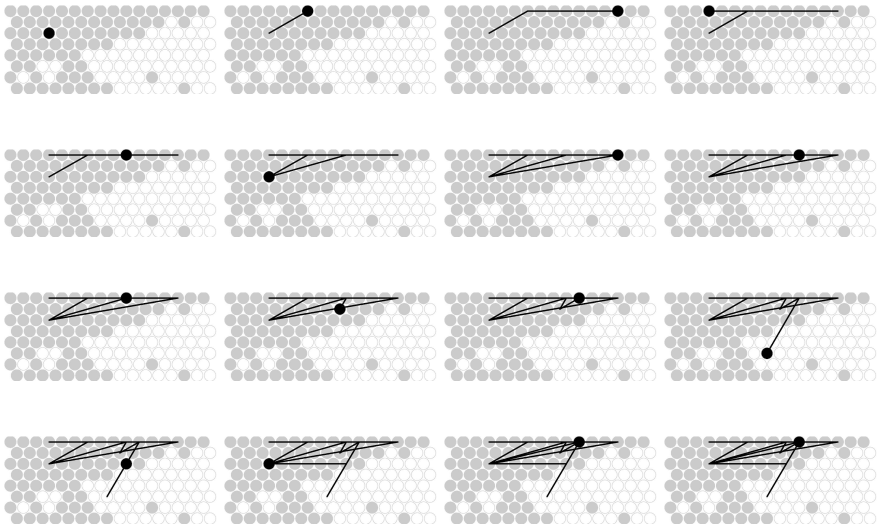

Fig. 5. Trajectory of the winning units for the *Divertimento in D (Mozart)*

4 Conclusions and future works

We have shown the ability of three different classifiers to map symbolic representations of melodies into a set of musical styles using melodic, harmonic and rhythmic descriptions. The best recognition rate has been found with a 7-descriptor model where syncopation, note duration, and pitch have an important role.

Some of the misclassifications can be caused by the lack of a smart method for melody segmentation. The music samples have been arbitrarily restricted to 8 bars, getting just fragments with no relation to musical motives. This fact can introduce artifacts in the descriptors leading to less quality mappings. The

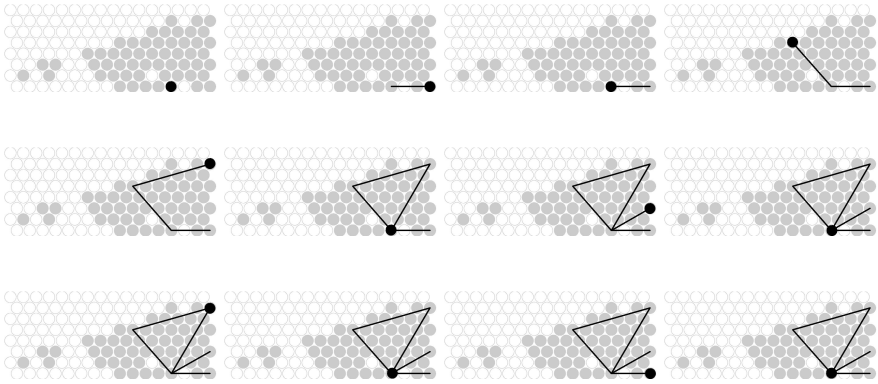

Fig. 6. Trajectory of the winning units for the jazz standard *Yesterdays*

main goal was to test the feasibility of the approach, and average recognition rates up to 88.7% have been achieved, that is very encouraging keeping in mind these limitations and others like the lack of valuable information for this task like timbre.

A number of possibilities are yet to be explored, like the development and study of new descriptors. It is very likely that the descriptor subset models are highly dependent on the styles to be discriminated. To achieve this goal a large music database has to be compiled and tested using our system for multiple different style recognition in order to draw significant conclusions.

The results suggest that the symbolic representation of music contains implicit information about style and encourage further research.

Acknowledgements

This paper has been funded by the Spanish CICyT project TAR, code: TIC2000–1703–CO3–02.

References

1. A. Rauber and M. Frühwirth. *Automatically analyzing and organizing music archives*, pages 4–8. 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2001). Springer, Darmstadt, Sep 2001.
2. Brian Whitman, Gary Flake, and Steve Lawrence. Artist detection in music with minnowmatch. In *Proceedings of the 2001 IEEE Workshop on Neural Networks for Signal Processing*, pages 559–568. Falmouth, Massachusetts, September 10–12 2001.
3. Hagen Soltau, Tanja Schultz, Martin Westphal, and Alex Waibel. Recognition of music types. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-1998)*. Seattle, Washington, May 1998.

4. O. A. S. Carpinteiro. A self-organizing map model for analysis of musical time series. In A. de Padua Braga and T. B. Ludermir, editors, *Proceedings 5th Brazilian Symposium on Neural Networks*, pages 140–5. IEEE Comput. Soc, 1998.
5. Belinda Thom. Unsupervised learning and interactive jazz/blues improvisation. In *Proceedings of the AAAI2000*, pages 652–657, 2000.
6. Petri Toiviainen and Tuomas Eerola. Method for comparative analysis of folk music based on musical feature extraction and neural networks. In *III International Conference on Cognitive Musicology*, pages 41–45, Jyvskyl, Finland, 2001.
7. Jeremy Pickens. A survey of feature selection techniques for music information retrieval. Technical report, Center for Intelligent Information Retrieval, Departament of Computer Science, University of Massachussetts, 2001.
8. Steven George Blackburn. *Content Based Retrieval and Navigation of Music Using Melodic Pitch Contours*. PhD thesis, Faculty of Engineering and Applied Science Department of Electronics and Computer Science, 2000.
9. Richard. O. Duda and Peter E. Hart. *Pattern classification and scene analysis*. John Wiley and Sons, 1973.
10. T. Kohonen. Self-organizing map. *Proceedings IEEE*, 78(9):1464–1480, 1990.
11. T. Kohonen, J. Hynninen, J. Kangas, and J. Laaksonen. Som_pak, the self-organizing map program package, v:3.1. Lab. of Computer and Information Science, Helsinki University of Technology, Finland, April, 1995. `http://www.cis.hut.fi/research/som_pak`.