Memory Based Automatic Music Transcription System for Percussive Pitched Instruments

Giovanni COSTANTINI[1,2], Massimiliano TODISCO[1], Renzo PERFETTI[3], Roberto BASILI[4], Daniele CASALI[1]
[1]Department of Electronic Engineering, University of Rome "Tor Vergata"
Rome, Italy
[2]Institute of Acoustics "O. M. Corbino", Via del Fosso del Cavaliere, 100
Rome, Italy
[3]Department of Electronic and Information Engineering, University of Perugia
Perugia, Italy
[4]Department of Computer Science, Systems and Production, University of Rome "Tor Vergata"
Rome, Italy

ABSTRACT

The target of our work dealt with the problem of extracting musical content or a symbolic representation of musical notes, commonly called musical score, from audio data of polyphonic music of percussive pitched instruments. We focuses on note events and their main characteristics: the onset (note attack instant) and the pitch (note name). Signal processing techniques based on the Constant-Q Transform (CQT) are used to create a time-frequency representation of the signal. The onset detection algorithm operates on a frame-by-frame basis and exploits a suitable time-frequency representation of the audio signal. The solution proposed consists of an onset detection algorithm based on Short-Time Fourier Transform (STFT), and a classification algorithm based on Support Vector Machine (SVM) to identify the note pitch. We introduce a memory based feature vector for classification. Moreover, to ascertain the effect of the memory, we evaluated the accuracy of the corresponding memoryless system. Finally, to validate our method, we present a collection of experiments using a wide number of musical pieces of heterogeneous styles, involving recordings of polyphonic music of three percussive pitched musical instruments.

Keywords: Music transcription, Onset detection, Constant-Q Transform, Support Vector Machine.

1. INTRODUCTION

Music transcription can be considered as one of the most demanding activities performed by our brain; not so many people are able to easily transcribe a musical score starting from audio listening, since the success of this operation depends on musical abilities, as well as on the knowledge of the mechanisms of sounds production, of musical theory and styles, and finally on musical experience and practice to listening.

The target of our work deals with the problem of extracting musical content or a symbolic representation of musical notes, commonly called musical score, from audio data of polyphonic music of percussive pitched instruments.

We must discern two cases in which the behaviour of the automatic transcription systems is different: monophonic music, where notes are played one-by-one and polyphonic music, where two or several notes can be played simultaneously.

Currently, automatic transcription of monophonic music is treated in time domain by means of zero-crossing or auto-correlation techniques and in frequency domain by means of Discrete Fourier Transform (DFT) or cepstrum. With these techniques, an excellent accuracy level has been achieved [1, 2].

Attempts in automatic transcription of polyphonic music have been much less successful; actually, the harmonic components of notes that simultaneously occur in polyphonic music significantly obfuscate automated transcription.

The first algorithms were developed by Moorer [3] Piszczalski e Galler [4]. Moorer (1975) used comb filters and autocorrelation in order to perform transcription of very restricted duets.

The most important works in this research field is the Ryynanen and Klapuri transcription system [5] and the Sonic project [6] developed by Marolt.

The solution proposed in this paper consists of an onset detection algorithm based on Short-Time Fourier Transform (STFT), and a classification algorithm to identify the note pitch.

The supervised classification method infers the correct note labels based only on training with tagged examples.

Polyphonic note transcription is obtained via a bank of Support Vector Machine (SVM) classifiers previously trained using, as spectral features, the result of Constant-Q Transform (CQT).

We introduce a short-term memory based feature vector for classification. Moreover, we examine the effect on transcription performance of different SVM kernels and different scales of amplitude spectrum values.

The paper is organized as follows: in the following section the onset detection algorithm will be described; in Section 3, the short-term memory spectral features will be formulated; Section 4 will be devoted to the description of the classification method; in Section 5, we will present the results of a series of experiments involving polyphonic piano, guitar and xylophone music. Some comments conclude the paper.

2. ONSET DETECTION

The aim of note onset detection is to find the starting time of each musical note. Several different methods have been proposed for performing this task [7, 8].

Our method is based on STFT and, notwithstanding its simplicity, it gives better or equal performance compared to

other methods [7, 8]. Let us consider a discrete time-domain signal s(n), whose STFT is given by

$$S_k(m) = \sum_{n=mh}^{mh+N-1} w(n-mh)s(n)e^{-j\Omega_N k(n-mh)} \qquad (1)$$

where N is the window size, ΩN = 2π/N, h is the hop size, m = 0, 1, 2,…, M is the hop number, k = 0, 1,…, N-1 is the frequency bin index, w(n) is a finite-length sliding Hanning window and n is the summation variable.

We obtain a time-frequency representation of the audio signal by means of spectral frames represented by the magnitude spectrum □Sk(m)□.

The values □Sk(m)□ can be packed as columns into a non-negative L×M matrix, where M is the total number of spectra we computed and L = N/2+1 is the number of their frequencies.

Afterwards, the rows of S are summed, and the following onset detection function, based on the first-order relative difference, is computed

$$f_{onset}(m) = \frac{f(m) - f(m-1)}{f(m)} \qquad (2)$$

where

$$f(m) = \sum_{l=1}^{L} S(l,m) \qquad (3)$$

The peaks of the function fonset can be assumed to represent the times of note onsets. After peak picking, a threshold T is used to suppress spurious peaks; its value is obtained through a validation process as explained in Section 5.

To demonstrate the performance of our onset detection method, let us show an example from real piano polyphonic music of Mozart's KV 333 Sonata in B-flat Major, Movement 3, sampled at 8 KHz and quantized with 16 bits.

We will consider the second and third bar at 120 metronome beat. It is shown in Figure 1.

We use a STFT with N = 512, an N-point Hanning window and hop size h = 256 corresponding to 32 milliseconds hop between successive frames. Figure 2 shows the onset detection function.

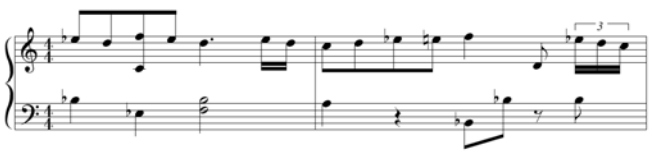

Figure 1. Musical score of Mozart's KV 333 Sonata in B-flat Major.

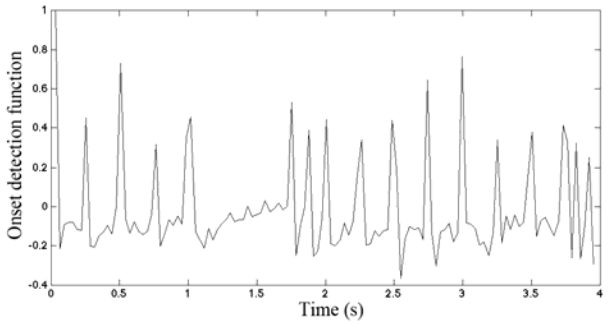

Figure 2. Onset detection function for the example in Figure 1.

3. THE CONSTANT-Q TRANSFORM AND THE SPECTRAL FEATURES

The Constant-Q Transform (CQT) [9] is similar to the Discrete Fourier Transform (DFT) with a main difference: it has a logarithmic frequency scale, since a variable width window is used. It suits better for musical notes, which are based on a logarithmic scale.

The logarithmic frequency scale provides a constant frequency-to-resolution ratio for every bin

$$Q = \frac{f_k}{f_{k+1} - f_k} = \frac{1}{2^{1/b} - 1} \qquad (4)$$

where b is the number of bins per octave and k the frequency bin. If b = 12, then k is equal to the MIDI note number (as in the equal-tempered 12-tone-per-octave scale). An efficient version of the CQT, based on the FFT and on some tricks, is presented in [10].

All the audio files that we used have a sampling rate of 8 kHz. The spectral resolution is b = 372, that means 31 CQT-bins per note, starting from note C0 (~ 32 Hz) up to note B6 (~ 3951 Hz). We obtain a spectral vector A composed by 2604 = 31 (CQT-bins) × 84 (musical notes).

To reduce the size of the spectral vector, we operate a simple amplitude spectrum summation among the CQT-bin relative to the fundamental frequency of the considered musical note, the previous 15 CQT-bins and the subsequent 15 CQT-bins; then, we obtain a spectral vector B composed by 84 = 1 (CQT-bins) × 84 (musical notes).

This can be formulated as follows

$$B(i) = \sum_{j=31\cdot i-30}^{31\cdot i} A(j) \qquad i = 1,2,..,84 \qquad (5)$$

Figure 3 shows the complete process of the spectral vector reduction.

Figure 4 shows the differences between three spectral vectors computed with b = 372 (4a), b = 84 (4b) and b = 372 with vector reduction (4c).

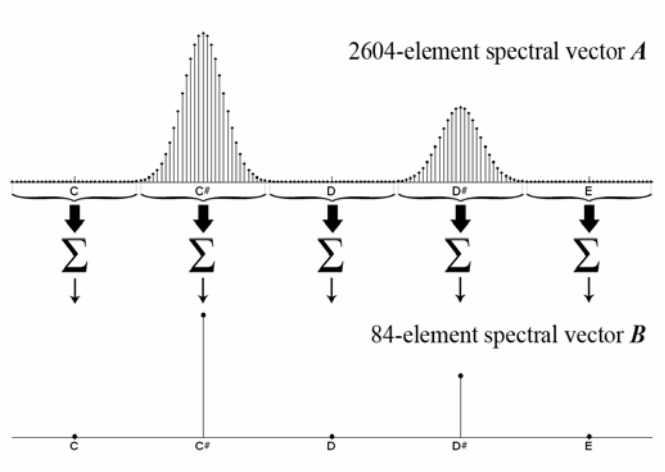

Figure 3. Reduction of the spectral vector.

Using (5) allows to obtain a greater accuracy in high frequency with the same vector length, as can be seen in Figures 4b and 4c.

The processing phase starts in correspondence to a note onset. Notice that two or more notes belong to the same onset if they are played within 32 ms. Firstly, the attack time of the note is discarded (in case of the piano, the longest attack time is equal to about 32 ms). Then, after Hanning windowing, a single CQT of the following 64ms is computed.

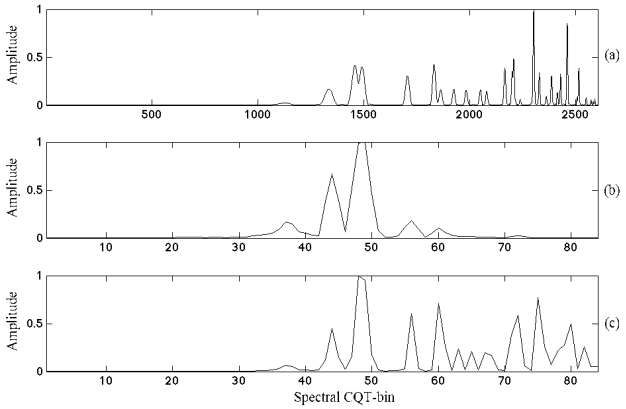

Figure 4. Spectral vectors of a polyphonic combination of note C3, G3 and B3 with b = 372 (a), b = 84 (b) and b = 372 with reduction (5) (c).

In our work, we take into account the following assumption: melodic and harmonic musical structures depend on the method adopted by the composer; this means that every musical note is highly correlated to the previous note in the composition.

Consequently, to improve classification results, we consider what happens before the onset, in particular, we introduce a short-term memory as follows: firstly, the segment of 32 ms preceding the onset time of the note is discarded, then, using Hanning windowing, the CQT on the previous 64 ms is computed. The output of the processing phase, including all the note onsets, is a matrix of 168 = 84 × 2 columns, corresponding to the CQT-bins, and a number of rows that is equal to the total number of note onsets in the Wave file computed with (2).

Two different feature vectors are considered for classification: they are based on two different scales of amplitude spectrum values, linear and logarithmic, rescaled into a range from 0 to 1.

4. MULTI-CLASS SVM CLASSIFICATION

A SVM identifies the optimal separating hyperplane (OSH) that maximizes the margin of separation between linearly separable points of two classes.

The data points which lie closest to the OSH are called support vectors. It can be shown that the solution with maximum margin corresponds to the best generalization ability [11].

Linearly non-separable data points in input space can be mapped into a higher dimensional (possibly infinite dimensional) feature space through a nonlinear mapping function, so that the images of data points become almost linearly separable.

The discriminant function of a SVM has the following expression

$$f(\mathbf{x}) = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \qquad (6)$$

where xi is a support vector, K(xi, x) is the kernel function representing the inner product between xi and x in feature space, coefficients αi and b are obtained by solving a quadratic optimization problem in dual form [11].

Usually, a soft-margin formulation is adopted where a certain amount of noise is tolerated in the training data. To this end, a user-defined constant C > 0 is introduced which controls the trade-off between the maximization of the margin and the minimization of classification errors on the training set [11].

The SVMs were implemented using the software SVMlight, developed by Joachims [12].

A linear kernel (5) and a radial basis function (RBF) kernel (6) were used

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j \qquad (7)$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right), \quad \gamma > 0 \qquad (8)$$

Linear SVMs need a regularization parameter C to be determined, while using the RBF kernel we need two parameters, C and γ. To this end we looked for the best parameter values in a specific range using a grid-search on a validation set. More details will be given in Section 5.

For multiclass classification, the one-versus-all (OVA) approach has been adopted. The OVA method exploits L SVMs, L being the number of classes. The ith SVM is trained using all the samples in the ith class with a positive class label and all the remaining samples with a negative class label.

Our transcription system uses 84 OVA SVM note classifiers whose input is represented by a 168-element feature vector, as described in Section 3.

The presence of a note in a given audio event is detected when the discriminant function of the corresponding SVM classifier is positive. Figure 6 shows a schematic view of the complete automatic transcription process.

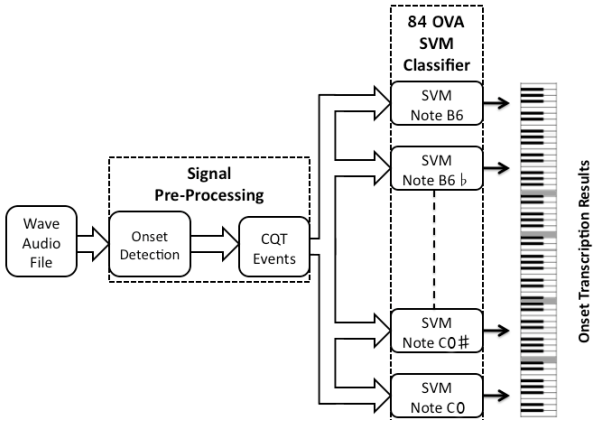

Figure 6. Schematic view of the complete automatic transcription process.

5. AUDIO DATASET AND EXPERIMENTAL RESULTS

In this section, we report the simulation results of our transcription system.

The MIDI data used in the experiments were collected from the Classical Piano MIDI Page, http://www.piano-midi.de/ [13]. A list of used pieces can be found in [13] (p. 8, Table 5).

The 124 pieces dataset was randomly split into 87 training, 24 testing, and 13 validation pieces. The first minute from each song in the dataset was selected for experiments, which provided us with a total of 87 minutes of training audio, 24 minutes of testing audio, and 13 minutes of audio for parameter tuning (validation set). This amounted to 22680, 6142, and 3406 note onsets in the training, testing, and validation sets, respectively.

First, we performed a statistical evaluation of the performance of the onset detection method. We consider as correct the onset detected within 32 ms of the ground-truth onset.

The results are summarized by three statistics: the Precision, the Recall and the F-measure, which are given by

$$Precision = \frac{TP}{TP + FP} \quad ; \quad Recall = \frac{TP}{TP + FN} \qquad (9)$$

$$Fmeasure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \qquad (10)$$

In the above formulas TP is the number of correct detections, FP is the number of false positives and FN is the number of false negatives. Precision represents the percentage of correct positive predictions in the identification of an example. Recall represents the capacity of the onset detector to identify the positive examples. The global variable F-measure is the harmonic mean of Precision and Recall.

As concerns the onset detection algorithm, we experimented with the threshold value to suppress spurious peaks. The reported results were obtained using the threshold value 0.02, 0.02, 0.05, referring to the three musical instruments; it was selected through maximization of the F-measure value on the 13 pieces of the validation dataset.

Table I quantifies the performance of the method on the test set (including 6142 onsets). The F-measure, outlined in Table I, can be compared with the results in [14], where a different onset function was used. The F-measure in [14] was 96.3% for piano, 94.8% for guitar and 95.6% for xylophone.

Table I

	Piano	**Xylophone**	**Guitar**
Precision	98.2%	96.8%	96.9%
Recall	96.2%	94.6%	95.7%
F-measure	**97.2%**	**95.7%**	**96.3%**

After detecting the note onsets, we trained the SVMs on the 87 pieces of the training set for each musical instrument, using both linear and logarithmic scale, and we tested the system on the 24 pieces of the test set. Moreover, to ascertain the effect of short-term memory, we evaluated the accuracy of the corresponding memoryless system, using the 84 CQT-bins feature vector, as described in Section 3.

The accuracy results are outlined in Table II, III and IV, referring to the three musical instruments.

Table II

Piano	**System WITH Short-term Memory**		**System WITHOUT Short-term Memory**	
	KERNEL		**KERNEL**	
SCALE	*Linear*	*RBF*	*Linear*	*RBF*
Linear	71.4%	72.7%	60.1%	65.3%
Logarithmic	75.4%	**80.3%**	68.9%	**73.5%**

Table III

Guitar	**System WITH Short-term Memory**		**System WITHOUT Short-term Memory**	
	KERNEL		**KERNEL**	
SCALE	*Linear*	*RBF*	*Linear*	*RBF*
Linear	70.8%	72.1%	59.7%	64.8%
Logarithmic	74.6%	**77.8%**	68.1%	**72.8%**

Table IV

Xylophone	**System WITH Short-term Memory**		**System WITHOUT Short-term Memory**	
	KERNEL		**KERNEL**	
SCALE	*Linear*	*RBF*	*Linear*	*RBF*
Linear	69.6%	70.4%	59.0%	63.5%
Logarithmic	71.6%	**76.9%**	67.0%	**71.5%**

In Tables II, III and IV, *Accuracy* denotes the accuracy metric proposed by Dixon [15], which is given by

$$Accuracy = \frac{TP}{TP + FP + FN} \qquad (11)$$

6. CONCLUSIONS

In this paper, we have discussed a polyphonic piano, guitar and xylophone transcription system based on the characterization of note events.

We focused our attention on temporal musical structure to detect notes. In particular, we considered a short-term memory preceding the note onset.

Different systems have been compared, based on feature vectors of 84 CQT-bins (memoryless) and 168 CQT-bins (with short-term memory), with linear or RBF kernel, and linear or logarithmic amplitude spectrum scale.

It has been shown that the proposed spectral reduction is helpful to lower computational cost without decreasing accuracy in the transcription system.

A wide number of musical pieces of heterogeneous styles were used to validate and test our transcription system.

A comparison of results shows the higher performance of the short-term memory based system with respect to the memoryless approaches.

9. REFERENCES

[1] J. C. Brown, "Musical fundamental frequency tracking using a pattern recognition method", **Journal of the Acoustical Society of America**, vol. 92, no. 3, 1992.

[2] J. C. Brown and B. Zhang, "Musical frequency tracking using the methods of conventional and narrowed autocorrelation", **Journal of the Acoustical Society of America**, vol. 89, no. 5, 1991.

[3] Moorer, "On the Transcription of Musical Sound by Computer". **Computer Music Journal**, Vol. 1, No. 4, Nov. 1977.

[4] M. Piszczalski and B. Galler, "Automatic Music Transcription", **Computer Music Journal**, Vol. 1, No. 4, Nov. 1977.

[5] M. Ryynanen and A. Klapuri, "Polyphonic music transcription using note event modeling," in **Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics** (WASPAA '05), New Paltz, NY, USA, October 2005.

[6] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," **IEEE Transactions on Multimedia**, vol. 6, no. 3, 2004.

[7] W.C. Lee, C.C. J. Kuo, "Musical onset detection based on adaptive linear prediction", **IEEE International Conference on Multimedia and Expo**, ICME 2006, Toronto, Canada, pp. 957-960, 2006.

[8] G.P. Nava, H. Tanaka, I. Ide, "A convolutional-kernel based approach for note onset detection in piano-solo audio signals", **Int. Symp. Musical Acoust.** ISMA 2004, Nara, Japan, pp. 289-292, 2004.

[9] J. C. Brown, "Calculation of a constant Q spectral transform", **Journal of the Acoustical Society of America**, vol. 89, no. 1, pp. 425–434, 1991.

[10] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant Q transform," **Journal of the Acoustical Society of America**, vol. 92, no. 5, pp. 2698–2701, 1992.

[11] J. Shawe-Taylor, N. Cristianini **An Introduction to Support Vector Machines**, Cambridge University Press (2000).

[12] T. Joachims, **Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning**, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[13] G. Poliner and D. Ellis, "A Discriminative Model for Polyphonic Piano Transcription", **EURASIP Journal of Advances in Signal Processing**, vol. 2007, Article ID 48317, pp. 1-9, 2007.

[14] G. Costantini, M. Todisco, R. Perfetti, "A Novel Sensor Interface for Detecting Musical Notes of Percussive Pitched Instruments", **Proceedings of IWASI IEEE International Workshop on Advances in Sensors and Interfaces**, Trani (Bari), Italy, June 25-26, 2009, pp. 121-126.

[15] S. Dixon, "On the computer recognition of solo piano music**",** in **Proceedings of Australasian Computer Music Conference**, pp. 31–37, Brisbane, Australia, July 2000.