On the Effect of Memory Width in Automatic Transcription Systems for Polyphonic Piano Music

Giovanni COSTANTINI[1,2], Massimiliano TODISCO[1], Giovanni SAGGIO[1]
[1]Department of Electronic Engineering, University of Rome "Tor Vergata", Rome, Italy
[2]Institute of Acoustics "O. M. Corbino", Rome, Italy

ABSTRACT

The objective of this study is to investigate the effect of memory in a transcription system for polyphonic piano music. The target of our work dealt with the problem of extracting musical content or a symbolic representation of musical notes, commonly called musical score. We focuses on temporal musical structures, note events and their main characteristics: the attack instant and the pitch and we compare the results obtained with four different feature vectors used in classification. In particular, we propose feature vectors based on one-event memory, two-events memory and three-events memory for classification. Moreover, we propose a supervised classification method that infers the correct note labels based only on training with labeled examples. The input to this system consists in piano music recordings stored in WAV files, while the pitch of all the notes in the corresponding score forms the output. The proposed system performs polyphonic transcription via a Support Vector Machine (SVM) classifiers, trained starting from spectral features obtained by means of the well-known Constant-Q Transform (CQT). Additionally, 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 piano.

Keywords: Piano music transcription, Memory, Constant-Q Transform, Support Vector Machine.

1. INTRODUCTION

Music transcription consists in transforming the musical content of audio data into a symbolic representation and it 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 piano.

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 a supervised classification algorithm to identify the note pitch. The supervised classification 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 feature vectors based on one-event memory, two-events memory and three-events memory for classification.

The paper is organized as follows: in the following section the spectral features will be formulated; Section 3 will be devoted to the description of the classification method; in Section 4, we will present the results of a series of experiments involving polyphonic piano music. Some comments conclude the paper.

2. THE CONSTANT-Q TRANSFORM AND THE SPECTRAL FEATURES

The Constant-Q Transform (CQT) [7] 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

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

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 [8].

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

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

Figure 1 shows the complete process of the spectral vector reduction.

Figure 2 shows the differences between three spectral vectors computed with b = 372 (2a), b = 84 (2b) and b = 372 with vector reduction (2c).

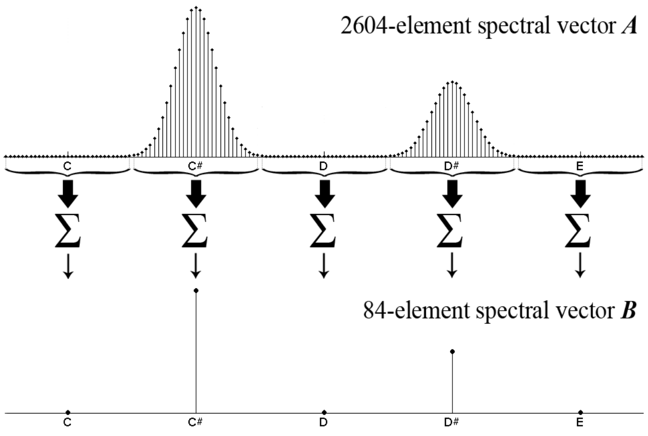
2604-element spectral vector *A*
C
C#
D
D#
E
Σ
84-element spectral vector *B*
C
C#
D
D#
E

Figure 1. Reduction of the spectral vector.

Using (2) allows to obtain a greater accuracy in high frequency with the same vector length, as can be seen in Figures 2b and 2c.

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 3 shows the complete process.

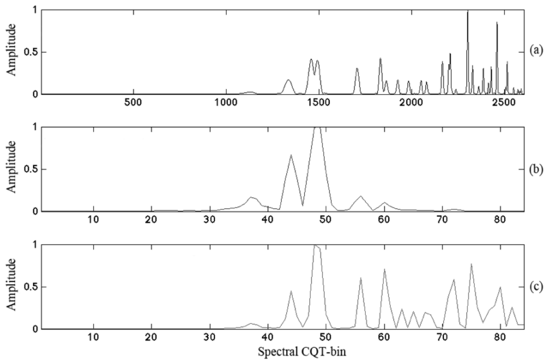

Figure 2. Spectral vectors of a polyphonic combination of note C3, G3 and B3 with b = 372 (a), b = 84 (b) and b = 372 with reduction (2) (c).

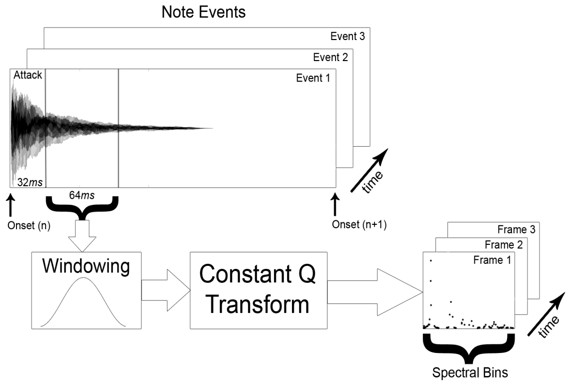

Figure 3. Feature extraction process

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, firstly we consider what happens before the onset at time *n*, in particular, we introduce one-event memory, two-events memory and three-events memory, this means that we consider what happens at onset time *n*-1, *n*-2 and *n*-3, respectively.

Figure 4 shows the spectral feature extraction, regarding one-event memory. 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.

Feature vectors are based on linear scales of amplitude spectrum values rescaled into a range from 0 to 1.

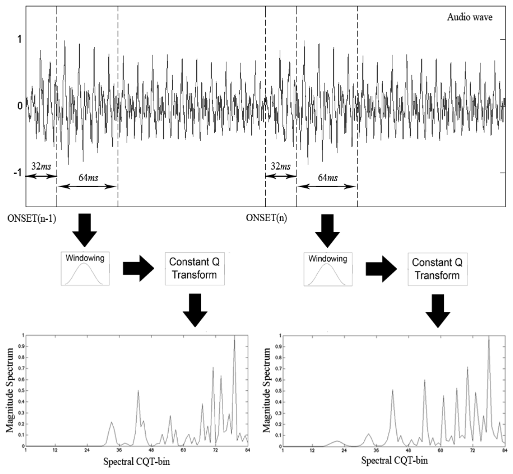

Figure 4. One-event memory.

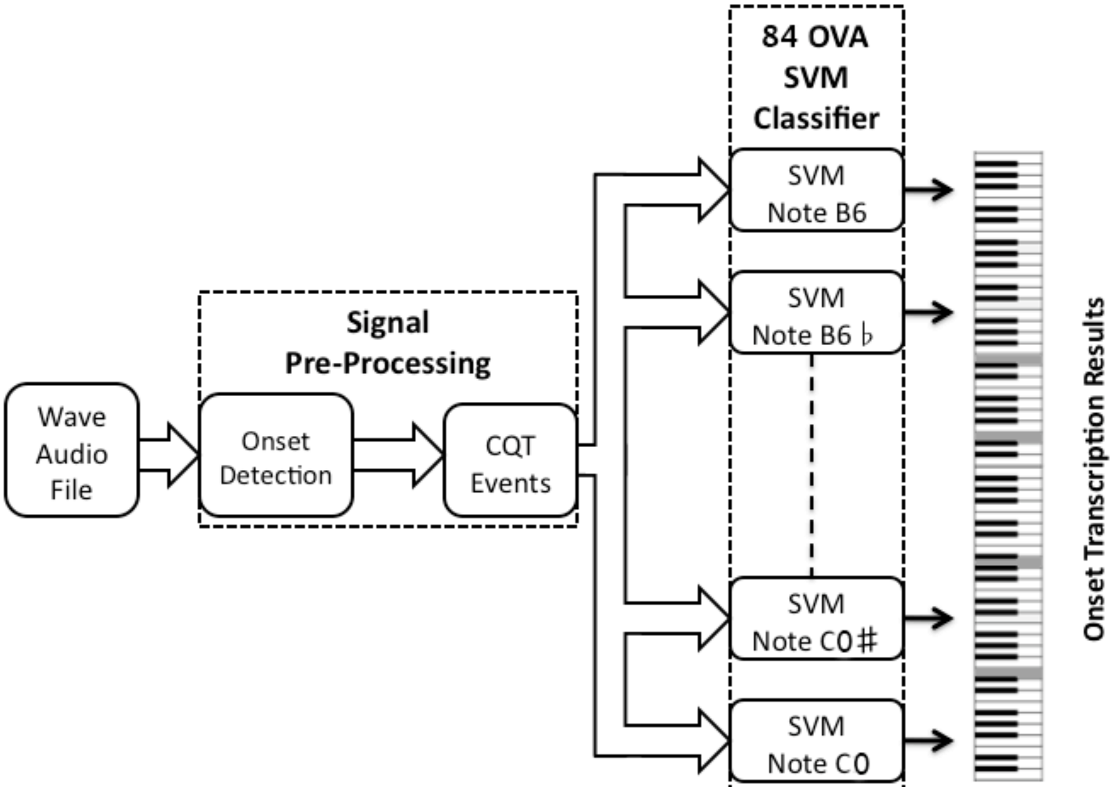

Figure 5. Schematic view of the complete automatic transcription process.

3. 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 [9].

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

$$f(\mathbf{x}) = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \qquad (3)$$

where x_i is a support vector, $K(x_i, x)$ is the kernel function representing the inner product between x_i and x in feature space, coefficients α_i and b are obtained by solving a quadratic optimization problem in dual form [9].

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 [9].

The SVMs were implemented using the software SVMlight, developed by Joachims [10].

A radial basis function (RBF) kernel were used

$$K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right), \quad \gamma > 0 \qquad (4)$$

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 4.

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 2.

The presence of a note in a given audio event is detected when the discriminant function of the corresponding SVM classifier is positive. Figure 5 shows a schematic view of the complete automatic transcription process.

4. 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. A list of used pieces can be found in [11] (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.

The results are summarized by the accuracy metric proposed by Dixon [12] which is given by

$$Accuracy = \frac{TP}{TP + FP + FN} \qquad (5)$$

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.

We trained the SVMs on the 87 pieces of the training set, using linear scale and we tested the system on the 24 pieces of the test set.

Moreover, to ascertain the effect of memory, we evaluated the accuracy of the corresponding memoryless system, using the 84 CQT-bins feature vector, as described in Section 2.

The accuracy results are outlined in Table I.

Table I

	Three-event Memory	**Two-event Memory**	**One-event Memory**	**Memoryless**
Acc (%)	**88.1**	**87.3**	**85.7**	**74.8**

5. CONCLUSIONS

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

We focused our attention on temporal musical structure to detect notes. In particular, we considered one-event memory, two-events memory and three-events memory for classification.

Different systems have been compared, based on feature vectors of 84 CQT-bins (memoryless) and 168 CQT-bins (with memory), with RBF kernel and linear 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 memory based system with respect to the memoryless approaches.

6. 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] 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.

[8] 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.

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

[10] 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.

[11] 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.

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