Transcription and Separation of Drum Signals From Polyphonic Music

Olivier Gillet, *Associate Member, IEEE*, and Gaël Richard, *Senior Member, IEEE*

***Abstract*—The purpose of this article is to present new advances in music transcription and source separation with a focus on drum signals. A complete drum transcription system is described, which combines information from the original music signal and a drum track enhanced version obtained by source separation. In addition to efficient fusion strategies to take into account these two complementary sources of information, the transcription system integrates a large set of features, optimally selected by feature selection. Concurrently, the problem of drum track extraction from polyphonic music is tackled both by proposing a novel approach based on harmonic/noise decomposition and time/frequency masking and by improving an existing Wiener filtering-based separation method. The separation and transcription techniques presented are thoroughly evaluated on a large public database of music signals. A transcription accuracy between 64.5% and 80.3% is obtained, depending on the drum instrument, for well-balanced mixes, and the efficiency of our drum separation algorithms is illustrated in a comprehensive benchmark.**

***Index Terms*—Drum signals, feature selection, harmonic/noise decomposition, music transcription, source separation, support vector machine (SVM), Wiener filtering.**

I. Introduction

THE development of the field of music information retrieval (MIR) has created a need for indexing systems that automatically extract semantic descriptions from music signals. This description would typically include melodic, tonal, timbral, and rhythmic information. So far, the scientific community has mostly focused on the extraction of melodic and tonal information (multipitch estimation, melody transcription, chords, and tonality recognition) but also to a lesser extent on the estimation of the main rhythmic structure. However, little effort has been made to obtain detailed information about the rhythmic accompaniment played by the drum kit in polyphonic music, despite the wide range of interesting applications that can be derived from its description. For instance, this information can ease genre identification, since many popular music genres are characterized by their distinct stereotypical drum patterns [1]. The rhythmic content can also be the basis of user queries, as illustrated by query by tapping or beatboxing systems [2], [3]. Finally, the availability of this description suggests new and interesting ways of playing and enjoying music—with applications such as drum track remixing or automatic DJing.

The problem of drum transcription has been initially addressed in the case of solo drums signals. Interested readers can refer to [4] for an extensive introduction to this topic and a review of existing systems. More recently, a variety of drum transcription systems have been developed to cope with signals in which the drums are played along with other instruments. All these systems follow one of these three approaches: *segment and classify*, *match and adapt*, or *separate and detect*. The first of these approaches consists in segmenting the signal into individual discrete events, and to classify each event using machine learning techniques. While this procedure has proved particularly successful on solo drum signals [5], [6], its application to polyphonic music [7]–[9] is more challenging, as most of the features used for classification are sensitive to the presence of background music. Efforts have been made lately by Paulus [10] to jointly perform the segmentation and the classification, as a single decoding process of a hidden Markov model. A second procedure consists in searching for occurrences of a reference temporal [11] or time–frequency [12] template within the music signal. A new template can be generated from the detected occurrences, and the matching/adaptation can subsequently be iterated. The last family of approaches relies on the intuition that the drum transcription process should simultaneously gain knowledge on the times at which drum instruments are played, and on their timbre. A possible way of achieving this is to decompose a time–frequency representation of the signal (such as its short-term Fourier transform) into a sum of independent components described by simple temporal and spectral profiles. The decomposition is traditionally achieved with independent subspace analysis (ISA) or nonnegative matrix factorization (NMF). In order to obtain components related to meaningful drum events, prior spectral profiles can be learned on solo drum signals and used to initialize the decomposition [13]. Alternatively, the decomposition can be performed with a fixed number of components, and heuristics [14] or classifiers [15] are used to identify, among the extracted components, those associated with each drum instrument to be transcribed. Such approaches highlight the links between music transcription and source separation, which aims to recover the signals of each individual musical source (instruments) from music recordings. Drum transcription could benefit from source separation techniques that would cancel the contribution of nonpercussive instruments from the signal. Reversely, the knowledge of the score could guide source separation.

The purpose of this article is to illustrate the relationships between transcription and source separation, in the context of

Manuscript received December 15, 2006; revised November 18, 2007. This work was supported in part by the European Commission under Contract FP6-027026-K-SPACE and in part by the National Project ANR-Musicdiscover. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mark Sandler.

O. Gilliet was with GET/Télécom Paris/CNRS LTCI, 75014 Paris, France. He is now with Google, Inc., CH-8001 Zurich, Switzerland. (e-mail: olivier.gillet@enst.fr).

G. Richard is with GET/Télécom Paris/CNRS LTCI, 75014 Paris, France (e-mail: gael.richard@enst.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2007.914120

drum signals. Our main contributions include the development of a complete drum transcription system which additionally uses a drum-enhanced version of the music signal to transcribe, the introduction of new (or the adaptation of existing) source separation techniques for the purpose of drum track extraction, and finally a thorough evaluation of the transcription and separation methods introduced by taking advantage of a large and fully annotated database of drum signals.

The outline of the paper is as follows. In Section II, we introduce two methods to enhance the drum track of a music signal. These methods can be considered as a first step toward a source separation algorithm. In Section III, a drum transcription system taking advantage of the drum-enhanced signal produced by these methods is presented. Section IV further investigates the problem of drum track separation. Section V summarizes some of our observations and suggests directions for future research. We finally present some conclusions in Section VI.

II. Drum Track Enhancement

This section describes two complementary techniques to enhance the drum track of polyphonic music signals. Such a processing can be included in music transcription systems (as done in Section III), or be considered as an elementary source separation algorithm. Both techniques use a similar decomposition of the signal into eight channels by an octave-band filter bank.[1]

A. Cancellation of Harmonic Sources From Stereo Signals

Most of the transcription and drum track extraction algorithms we reviewed only consider monophonic (single channel) signals. However, the recordings of popular music produced in the last few decades are mostly stereophonic signals. Traditionally, the left and right channels of such recordings are simply averaged before further processing. It would nevertheless be more optimal to recover as much of the drum signal as possible from the stereophonic mix.

Our approach, specific to drums, is based on the same assumptions and motivations as ADRess [17]: First, most popular music is produced using to the so-called *Panoramic* mixing—the left and right channels being linear combinations of monophonic sources. Second, we observed that some instruments in the mix are more predominant in some frequency bands than others. That is to say, in a narrow frequency band, the signal can be considered as a mixture of a predominant instrument, panned at a given position, and remaining components spread across the stereo field.

The stereo signal is consequently split into eight subbands, by means of the filter bank previously described. An independent component analysis (ICA) is applied to each pair of subband stereo signals, resulting in the extraction of two pseudosources and an unmixing matrix per subband. A support vector machine (SVM) classifier is trained to discriminate, among the extracted pseudosources, those containing drum sounds, and those containing only harmonic instruments. For this purpose, the amplitude envelope of each subband pseudosource is computed,[2] and the temporal features described in [15] are extracted. The subband index is used as an additional feature, since some subbands are more likely than others to contain percussive pseudosources. The output signal is synthesized by applying a null gain to all the subband pseudosources which are identified as containing no drums.

The SVM is trained on a subset of files unrelated to the evaluation database, and gives an estimate of the posterior probability $p(y|\mathbf{x})$, where y is the class (percussive/nonpercussive) and $\mathbf{x}$ the extracted features. This classifier is likely to commit two kinds of misclassification errors: nonpercussive instruments can be classified as drum sources and kept in the mix, and drum instruments can be classified as nonpercussive sources and suppressed from the mix. The former type of error is more preferable than the latter for the task at hand. Assuming the cost of not including a percussive source in the mix is twice the cost of including a nonpercussive source, the optimal decision function is $p(y|\mathbf{x}) > (1/3)$. The ability of this method to separate the drums from stereo signals is tested in Section IV-C, and has already shown interesting results. For instance, predominant instruments such as electric bass or organ could be removed efficiently from some subbands of the signal. Nevertheless, due to the bias introduced in the classification, this method left some of our test signals unchanged.

B. Bandwise Harmonic/Noise Decomposition

The principle of this approach is to decompose each subband signal into stochastic and harmonic components. Because unpitched percussive sounds (in particular the hi-hat and snare drum) have mostly nonharmonic components located in well-defined subbands, and because the other melodic instruments have mostly harmonic components, the extracted stochastic components essentially contain the contribution of the drums.

1) Harmonic/Noise Decomposition: This step aims to decompose the (real valued) subband signals s into a harmonic part h, modeled as a sum of M exponentially damped sinusoids [18], and a noise residual r. While traditional Fourier analysis could be used to detect sinusoidal components, its temporal and frequency resolution cannot be adjusted independently. More promising results are achieved by subspace-based methods, the principle of which is briefly exposed here. Let us consider the $L \times L$ Hankel data matrix formed from a signal window $[s(0), \ldots, s(2L-2)]$

$$\mathbf{H} = \begin{bmatrix} s(0) & s(1) & \ldots & s(L-1) \\ s(1) & s(2) & \ldots & s(L) \\ \vdots & \vdots & \ddots & \vdots \\ s(L-1) & s(L) & \ldots & s(2L-2) \end{bmatrix}.$$

Its eigenvalue decomposition (EVD) yields $\mathbf{H} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$. Let us call $\mathbf{U}_s$ the matrix formed by the $2M$ columns of $\mathbf{U}$ associated to the eigenvalues with the highest magnitudes. It can be demonstrated [18] that the harmonic part of the signal, modeled as a sum of M exponentially damped sinusoids, belongs to the $2M$-dimensional space of which $\mathbf{U}_s$ is a basis. This har-

[1]Uniform and logarithmic (octave-band) filter banks, followed by harmonic/noise decomposition, have been compared in [16] for the purpose of note onset detection. In our case, because discriminating snare drum and bass drum events requires a higher frequency resolution in the lowest frequency band, the octave-band filter bank is preferred.

[2]This envelope is estimated as $|x + j\mathcal{H}(x)| * h$, where $\mathcal{H}$ is the Hilbert transform and h a 100-ms-long half Hann window.

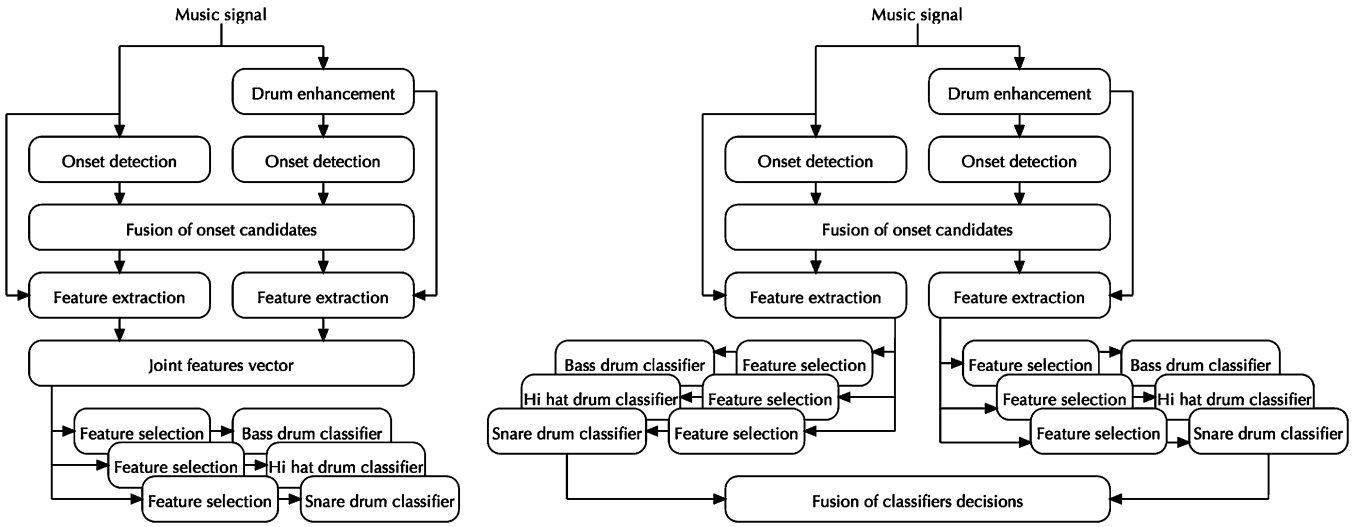

Fig. 1. Overview of the transcription system, illustrating the two fusion methods: early fusion (left), and late fusion (right).

monic part can thus be obtained by projection onto this subspace according to $\mathbf{h} = \mathbf{U}_s\mathbf{U}_s^H\mathbf{s}$, where $\mathbf{s} = [s(0), \ldots, s(L-1)]$ and $\mathbf{h} = [h(0), \ldots, h(L-1)]$. The noise subspace is defined as the $L - 2M$ dimensional orthogonal complement to $\mathbf{U}_s$. The stochastic part is extracted similarly by projection onto the noise subspace: $\mathbf{r} = \left(\mathbf{I} - \mathbf{U}_s\mathbf{U}_s^H\right)\mathbf{s}$. The EVD being computationally expensive, the matrix $\mathbf{U}_s$ is updated for each new observation window using the sequential iteration algorithm [18].

This noise subspace projection is applied to each of the subband signals produced by the filter bank previously described. This considerably reduces the computational load of the decomposition. The window size used for the kth subband signal was $\max(2^{k+2}, 32)$. This ensured, for the lowest bands, that $L \gg 2M$, while taking into account the fact that each subband has been increasingly decimated. The number of sinusoids to extract per band has been fixed to, respectively, 2, 4, 6, and 6 in the first four bands, 12 in the remaining bands—except for the last band which is not processed and considered as entirely stochastic.[3] These results can be compared to the observations reported in [16]—our numbers are slightly lower so as to avoid overestimation of the number of sinusoids especially in the lower band where harmonic components of the bass drum are often present.

2) Usefulness of the Decomposition: The full-band drum-enhanced signal is obtained by synthesis from the stochastic components of each subband signals. Clearly, nonpercussive instruments are strongly attenuated in this synthesized signal. In fact, it will be shown in Section III that the combination of the stereo harmonic source cancellation described in Section II-A with this noise subspace projection is an efficient preprocessing algorithm for drum track transcription. Nevertheless, it should be underlined that this simple resynthesis is not efficient for high-quality drum track separation. First, nonpercussive instruments may also have a stochastic component (e.g., breath for wind instruments, hammer strike for piano) which needs to be eliminated from the separated signal. Second, the bass drum and snare drum have harmonic components which should not be eliminated. An improved synthesis for high-quality source separation applications will thus be presented in Section IV.

[3]Order estimation techniques (such as [19] for example) can be used to estimate the number of sinusoids per band. However, in our context, it was found that frequent changes of the model order with time was more detrimental to the quality than a fixed well chosen order for each band.

III. Drum Transcription From Polyphonic Music

A. Overview

The drum transcription system described in this article follows the *segment and classify* approach: salient events, which may be drum events, are detected from the music signal. A set of features is extracted in the neighborhood of each note onset. The actual recognition of drum events is performed by multiple binary classifiers, each of them trained to detect the presence of a target instrument of the drum kit (bass drum, snare drum, etc.). In this paper, we focus on bass drum (BD), snare drum (SD), and hi-hat (HH) detection, since the most typical and recognizable rhythmic patterns used in popular music are played on these instruments.

A specificity of our work is that the original music signal is processed by the drum enhancement algorithm described above, which aims to amplify or extract the drum track. Then, onset detection and feature extraction are simultaneously performed on the original and drum-enhanced signals. This choice is motivated by the following observation: On the one hand, some of the features extracted from the original music signal are very sensitive to the presence of the other instruments in the mix (for example, the spectral centroid might be shifted toward the higher frequencies when a high-pitched note is played along with a bass-drum hit). On the other hand, the features extracted from the drum-enhanced signal are noisier due to the artifacts introduced by the drum enhancement process. Thus, our approach aims to combine both feature sets to gain robustness. This combination can be achieved either by early fusion, where features extracted from each signal are merged into a single feature vector which is then processed by a set of classifiers; or late fusion, where a different set of classifiers and features is used for both signals, and where the decisions of these classifiers are aggregated to yield the final transcription. The overall transcription process is described for both cases in Fig. 1, and each component is presented in detail below.

It is worth mentioning that the *segment and classify* approach is suitable for near real-time applications as it is nearly causal (a 200-ms lookahead is required for feature extraction) and computationally inexpensive since onset detection, feature extraction and classification can be performed in less than real-time on common personal computers. The harmonic/noise decomposition presented in Section II-B introduces an additional algorithmic delay of 277 ms. In contrast, the preprocessing of stereo signals introduced in Section II-A is not causal, though computationally inexpensive.

B. Onset Detection

The detection of salient events is performed by means of the note onset detection algorithm described in [20]. This algorithm splits the signal into 512 frequency channels using a short-term Fourier transform (STFT), yielding the spectrogram $\hat{X}(m,k)$. In each frequency band k, the signal $|\hat{X}(m,k)|$ is low-pass filtered, and its dynamic range is compressed to produce a perceptually plausible power envelope. Then, its derivative is computed by applying an optimal finite-impule response (FIR) differentiation filter, resulting in the spectral energy flux. A detection function $d(m)$, which exhibits sharp peaks at the onset of notes or drum hits, is obtained by summing the spectral energy flux across all frequency bins. A median filter is applied to the detection function to define a dynamic threshold function $\tau(m)$, and a note onset is detected whenever $d(m) > \alpha\tau(m)$.

In our work, the detection functions are separately computed for the original and for the drum-enhanced signal. The two detection functions are then summed to obtain a common set of onsets which will subsequently be used for feature extraction.[4]

The parameters of the onset detector are adjusted to favor a high recall rate (at the cost of a lower precision rate). In fact, detecting onsets associated to other instruments is not troublesome, since such events can be discarded later at the classification stage.

C. Feature Extraction

There is no consensus on the most relevant feature set for discriminating several classes of unpitched drum instruments. A large variety of descriptors are used in the different studies, sometimes associated with statistical feature selection (see, for example, [6] and [21]). It is not clear if these choices are still relevant for the classification of unpitched drum instruments in the presence of background music. More recently, Tanghe *et al.* have described a classification system in [7] which uses computationally inexpensive temporal and spectral features, along with Mel frequency cepstral coefficients (MFCCs). Though it is not exactly a feature selection process, the parameters of the MFCCs extractor have been optimized by means of simulated annealing in [22]. Some of these features have a direct perceptual or acoustical interpretation (for instance, MFCCs capture the shape of the spectral envelope) which justifies their use for the task at hand. While some other features might not have such interpretations, they can have a significant discriminative power. In this paper, we decided to emphasize on the classification efficiency of features rather than on their perceptual or acoustic meaning. We consequently examine a large subset of candidate features and select the most relevant ones using machine learning techniques. This approach, which trades interpretability for classification efficiency, was successfully applied to musical instrument recognition by Essid *et al.* in [23].

Similarly, the duration of the observation windows on which the features are computed greatly varies amongst studies. It ranges from fixed 80-ms-long windows starting at each observed onset [7] to windows defined between two tatum[5] grid points [5]. In [6], we used the entire interval between two consecutive strokes as an observation window. This choice makes the feature extraction process more robust—since, for example, the estimation of the spectrum or amplitude envelope benefits from the large number of available samples—but introduces variability, as the same feature might be computed on only the attack of a stroke, or on its entire duration. To ensure the robustness of the extracted features, while minimizing the variability of the extracted features, we decided to use as many samples as possible, within a 200-ms time frame. Hence, the features associated to the onset t_i are computed on the window $[t_i, \min(t_i + 0.2, t_{i+1})]$.

The 147 features considered in this work are shortly presented here.

- **Temporal features (6)**: These features include the crest factor, temporal centroid, and the zero-crossing rate—computed in both its standard and noise-robust version [7]. Additionally, an exponential decay Ae^{-Bn} is fitted to the amplitude envelope of the signal (see note 2), the parameters A and B being used as features.
- **Energy distribution features (25)**, which include the following.
  - — **Overall signal energy** computed as the logarithm of the root mean square (lRMS) of the signal across the entire observation window.
  - — **Energy of the output of matched filters** computed as the lRMS of the output of three filters adapted to the frequency content of the bass drum, snare drum, and hi-hat signals [7]. Additionally, the lRMS difference between adjacent frequency bands, as well as the difference between the lRMS in each band and the lRMS of the original signal is measured.
  - — **Energy in drum-specific filter bank** obtained as the lRMS of the signal in each band of the filter bank described in [6].
  - — **Energy ratio in an octave-band filter bank** obtained as the difference of lRMS between adjacent bands of a bank of overlapping octave-band filters (see [23]).
- **Spectral features (12):** They include the four spectral moments, the spectral rolloff and flatness (see [24]), and the first six linear prediction coefficients, which are a rough estimate of the spectral envelope.
- **Cepstral features (78)**: They consist of the average and standard deviation of the 13 first MFCCs, ΔMFCCs, and Δ^2MFCCs across the observation window.

[4]Different methods and operators (such as product, minimum or maximum) were tested for combining the detection functions. The results obtained were all very similar which may be due to the fact that our drum-enhancement method preserves transients from nonpercussive instruments well, and that more generally, the selected onset detection algorithm performs particularly well on sharp, impulsive signals, such as drum hits.

[5]The tatum is a subdivision of the main tempo and refers to the smallest common rhythmic pulse.

- **Perceptual features (26)**: The relative specific loudness, sharpness, and spread (Sp) are computed, according to their description given in [24].

To obtain centered and unit variance features, a linear transformation is applied to each computed feature. This normalization scheme is more robust to outliers than a mapping of each feature's dynamic range to $[-1, 1]$.

D. Feature Selection

Training a classifier on the large feature set extracted above is intractable, as some of the extracted features can be noisy, redundant with others, or unable to discriminate the target classes. The goal of feature selection is to avoid such problems by selecting the subset of the *most efficient* d features. This issue has been addressed extensively in the machine learning community (see [25] for an introduction to the topic). Features can be selected according to three categories of algorithms. *Wrapper* algorithms [26] assess the usefulness of a candidate feature set by evaluating its performance for the subsequent classification step. The resulting feature set consequently depends on the machine learning algorithm selected for the classification step, making it prone to overfitting [27]. Oppositely, *filter* algorithms do not require the choice of a classification method. Such methods measure the relevancy of each feature according to two criteria: redundancy of this feature with respect to the others, by means of similarity measures [28], and discriminative power of the feature with respect to the known class labels. Finally, *embedded* algorithms consider the decision function produced by a classifier to gain knowledge on the weight or relevance of each feature [29]. In this paper, we evaluated both a *filter* and an *embedded* feature selection strategy.

Inertia Ratio Maximization Using Feature Space Projection (IRMFSP):

In the context of a binary classification problem, let N^+ and N^- be the number of positive and negative examples, $N = N^+ + N^-$ the total number of training examples, $\mathbf{x}_k^+$ (resp. $\mathbf{x}_k^-$) the kth feature vector from the positive (resp. negative) class, and $\mathbf{m}^+$ (resp. $\mathbf{m}^-$) the means of feature vectors from the positive (resp. negative) class. The Fisher criterion can be defined as

$$r = \frac{\frac{N^+}{N}\|\mathbf{m}^+ - \mathbf{m}\| + \frac{N^-}{N}\|\mathbf{m}^- - \mathbf{m}\|}{\frac{1}{N^+}\sum_{k=1}^{N^+}\left\|\mathbf{x}_k^+ - \mathbf{m}^+\right\| + \frac{1}{N^-}\sum_{k=1}^{N^-}\left\|\mathbf{x}_k^- - \mathbf{m}^-\right\|}.$$

Intuitively, it measures the ratio between the inter-class and intra-class scatter, a large value of r ensuring a good discrimination between classes. Thus, the IRMFSP algorithm [30] iteratively builds a feature set $\mathbf{X}_n$ according to two steps.

1) Relevancy maximization: The feature maximizing the Fisher discriminant r_{i+1} is selected and appended to $\mathbf{X}_n$, yielding a new subset $\mathbf{X}_{n+1}$.
2) Redundancy elimination by orthogonalization: The remaining features are obtained by subtraction of their projection on the space spanned by the already selected features.

To obtain a ranking of the features, this process is continued until n reaches the total number of features.

Recursive Feature Elimination With Support Vector Machines (RFE-SVM):

The RFE-SVM algorithm [29] iteratively removes from the entire feature set those features whose contribution to the decision function of a linear SVM is minimal.

1) A linear SVM is trained on the surviving feature set X_n, yielding a decision function $y_n(\mathbf{x}) = \sum_{k=1}^{N} \alpha_k \mathbf{x} \cdot \mathbf{x}_{nk}$, where the α_k are Lagrange multipliers, and $\mathbf{x}_{nk}$ the training examples, using only the features selected in X_n.
2) The weight of the jth feature is computed as $w(j) = \left(\sum_{j=1}^{N} \alpha_j \mathbf{x}_{nk}(j)\right)^2$ where $\mathbf{x}_{nk}(j)$ is the jth component of $\mathbf{x}_{nk}$.
3) The feature(s) with the smallest weight is(are) removed, yielding a new surviving feature set X_{n+1}.

Since training the SVM can be computationally expensive, a large number of features can simultaneously be eliminated during the first iterations. In the following experiments, 25% of the surviving features are eliminated at each iteration, until less than 32 features remain. Afterward, the features are eliminated one by one.

Both algorithms were used to obtain a ranking of the most relevant features. The final number of features retained was selected by a grid search from the set $\mathcal{D}(d) = \{4, 8, 16, 32, 64, 96\}$. We found that RFE-SVM performed better than IRMFSP except for small feature sets (less than eight features). Thus, in the rest of this paper, IRMFSP is used for feature selection when $d \in \{4, 8\}$, and RFE-SVM is used in the other cases.

E. Classification

We aim to assign the set of instruments of the drum kit played at t_i to each feature vector $\mathbf{x}_i$ extracted at time t_i. Considering a subset of K instruments of the kit (in our case, $K = \{\text{bass drum}, \text{snare drum}, \text{hi-hat}\}$), $2^{|K|}$ combinations of instruments are possible, including the combination $\emptyset$ where no rhythmic instrument is played. Such a classification problem can be solved by either a $2^{|K|}$-class classifier or by $|K|$ binary classifiers—each of them detecting the presence or absence of a target instrument in K. The former strategy leads to homogenous classes in unbalanced proportions. The latter solution, which is used for the rest of this paper, yields less homogenous classes (for example, the positive examples for the bass drum detector will include both bass drum strokes and bass drum + snare drum combinations), but the number of positive and negative training examples is more balanced for each classifier. Refer to [6] for an experimental comparison of the two strategies.

The classifiers selected for this task are C-support vector machines (C-SVM), whose generalization properties and discriminative power have been proved on a wide range of tasks, and for which efficient software implementations are available. Interested readers can refer to [31] or to [32] for a more theoretical presentation of the underlying structural risk minimization theory. A normalized Gaussian kernel $K(\mathbf{x}, \mathbf{y}) = \exp\left(\|\mathbf{x} - \mathbf{y}\|^2 / 2d\sigma^2\right)$ (where d is the number of features) is chosen to allow for nonlinear decision boundaries.

A grid search was used to determine the parameters σ and C, both expressing the tradeoff between misclassification and generalization errors—the candidate values of these parameters being $\mathcal{D}(\sigma) = \{(1/2), 1, 2, 4\}$ and $\mathcal{D}(C) = \{2, 16, 128, 1024\}$. Finally, a sigmoid function is fit to the decision function of the SVM, according to the method described by Platt in [33], to obtain posterior probabilities of class membership rather than "hard" decisions from the classifiers—this allows for the adjustment of a decision threshold, to reach an acceptable precision/recall tradeoff, or for further information fusion.

F. Information Fusion

As described in Section III-A, two fusion schemes are considered to take into account the original and drum enhanced signals in the classification. *Early fusion* consists in joining the two feature vectors obtained from both sources and applying the feature selection and classification process to this large vector. *Late fusion* employs two different sets of classifiers for each feature set, and then aggregates the posterior probabilities given by each classifier. A variety of aggregation operators were considered, such as the product, sum, maximum, minimum, weighted norms [34], and a "most confident" operator defined as

$$F(p_1, p_2) = \begin{cases} p_1, & \text{if } |p_1 - 0.5| > |p_2 - 0.5| \\ p_2, & \text{otherwise.} \end{cases}$$

Best classification results are obtained with the sum and maximum operators.

G. Evaluation Protocol

1) Experimental Database: Our experiments were conducted on the *minus one* sequences of the ENST-drums database [35]. These sequences are based on 17 instrumental songs without drums, of an average length of 71 s, for which three different drummers performed a drum accompaniment. An interesting characteristic of this material is that the mixing between the drums and the musical accompaniment can be freely adjusted in order to assess the robustness of the transcription algorithm in the presence of background music. The experiments described in Section III-G2 are repeated on four mixes, in which the background accompaniment is respectively suppressed, attenuated by 6 dB, balanced with the drum, and amplified by 6 dB. This database can be considered difficult as far as the drum playing style is concerned: some of the sequences are played with brushes or mallets; some others emphasize on a rich and natural drum playing style. In particular, ghost notes, which are de-emphasized strokes used to give a feeling of "groove," are included in the annotation and are particularly challenging to detect.

2) Protocol: In the evaluation, care has been taken to avoid overfitting and excessive fine-tuning of classification parameters. To this purpose, the 17 songs of the database are divided into three groups (one group contains the five longest songs, the two other groups contain six songs each). Let S_{ij} be the subset of the database containing the songs from the ith group played by drummer j. Our evaluation protocol is a *nested cross-validation* described by the pseudocode in III-G2 and illustrated in Fig. 2.

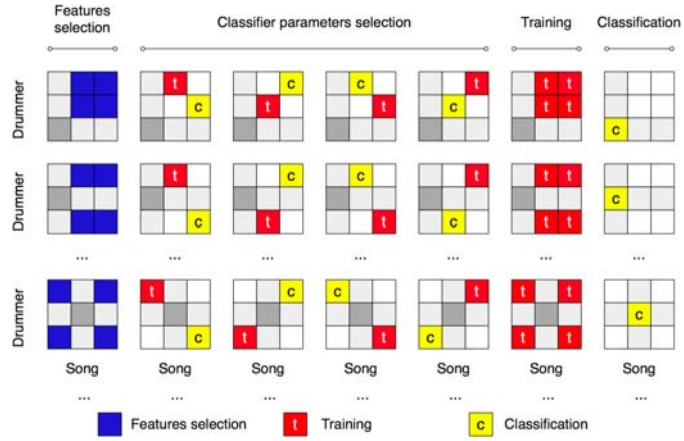

Fig. 2. Nested cross-validation protocol.

Algorithm 1 Evaluation protocol

Input: Database split in 9 groups S_{ij}, extracted features
  for all $(i_0, j_0) \in \{1, 2, 3\}^2$ **do**
    for all Binary instrument classification problem **do**
      Rank the features in the subset $\cup_{i \neq i_0, j \neq j_0} S_{ij}$
      for all $(C, \sigma, d) \in \mathcal{D}(C) \times \mathcal{D}(\sigma) \times \mathcal{D}(d)$ **do**
        error $\leftarrow 0$
        for all $i_1 \neq i_0, j_1 \neq j_0$ **do**
          Train a C-SVM using parameters C, σ and d best features, on $S_{i_1 j_1}$
          Test this classifier on $S_{i_2 j_2}$, where $i_2 \notin \{i_0, i_1\}, j_2 \notin \{j_0, j_1\}$
          error $\leftarrow$ error + classification error
        end for
      end for
      Train a C-SVM using the parameters C^*, σ^* and d^* best features on $S_{i_1 j_1}$, where C^*, σ^* and d^* minimize the generalization error
    end for
    Use the binary classifiers to label the data from $S_{i_0 j_0}$
  end for
Output: An automatic transcription for each sequence of the entire database

This protocol ensures that the selected parameters for C, σ, and the number of features d provide good generalization properties—since in the inner loop of our protocol, the training and testing sets correspond to both different songs and drummers. Overfitting is prevented by ensuring that the data on which the classifiers will ultimately be tested have nothing in common with the data on which the features and the classification parameters are optimized.

3) Evaluation Metrics: The accuracy of the automatic transcription is evaluated by standard precision and recall scores, computed for each target instrument class k, and by the F-measure, which summarizes the tradeoff between precision and recall. Let N_d be the total number of strokes of instrument k detected by the system, N_c the number of correct strokes detected by the system (a deviation of up to 50 ms being allowed between actual and detected drum events), and N the actual number of

TABLE I
DRUM TRANSCRIPTION ACCURACY FOR VARIOUS BACKGROUND MUSIC LEVELS, ON ALL THE *MINUS* ONE SEQUENCES OF THE ENST-DRUMS DATABASE

	Original signal only			Drum-enhanced signal			Early fusion			Late fusion		
Instrument	Recall	Prec.	F meas.	Recall	Prec.	F meas.	Recall	Prec.	F meas.	Recall	Prec.	F
Accompaniment $\mp$ dB												
BD	66.4%	67.8%	67.1%	60.4%	75.2%	67.0%	62.8%	62.7%	62.8%	**65.6%**	**80.5%**	**72.3%**
SD	52.4%	80.1%	63.3%	57.0%	70.1%	62.9%	51.1%	78.3%	61.8%	**58.5%**	**75.7%**	**66.0%**
HH	81.3%	76.8%	79.0%	82.5%	78.6%	80.5%	86.5%	76.6%	81.3%	**85.2%**	**79.2%**	**82.1%**
Accompaniment 6 dB												
BD	65.7%	72.1%	68.7%	54.3%	69.3%	60.9%	63.7%	61.5%	62.6%	**64.6%**	**79.2%**	**71.1%**
SD	54.7%	72.4%	62.3%	57.3%	69.0%	62.6%	56.6%	75.1%	64.5%	**57.7%**	**73.2%**	**64.5%**
HH	81.2%	75.8%	78.4%	79.5%	78.4%	79.0%	80.5%	77.3%	78.9%	**82.4%**	**78.2%**	**80.3%**
Accompaniment +0 dB												
BD	61.7%	58.4%	60.0%	54.1%	65.8%	59.4%	61.1%	61.0%	61.1%	**62.0%**	**70.2%**	**65.8%**
SD	46.4%	66.7%	54.7%	50.6%	66.1%	57.4%	52.0%	69.5%	59.5%	**50.6%**	**70.7%**	**59.0%**
HH	80.8%	70.6%	75.4%	79.5%	73.3%	76.3%	78.9%	74.9%	76.8%	**83.1%**	**73.0%**	**77.7%**
Accompaniment +6 dB												
BD	60.0%	54.3%	57.0%	55.1%	58.5%	56.8%	55.5%	54.9%	55.2%	**60.9%**	**62.6%**	**61.7%**
SD	37.6%	54.7%	44.6%	41.3%	56.5%	47.7%	**48.0%**	**58.7%**	**52.8%**	42.8%	60.4%	50.1%
HH	76.7%	65.6%	70.6%	74.7%	68.4%	71.4%	74.7%	67.7%	71.1%	**78.0%**	**68.0%**	**72.6%**

TABLE II
DRUM TRANSCRIPTION ACCURACY ON THE *MINUS* ONE SEQUENCES OF THE PUBLIC SUBSET OF THE ENST-DRUMS DATABASE

	Original signal only			Drum-enh anced signal			Early fusion			Late fusion		
Instrument	Recall	Prec.	F meas.	Recall	Prec.	F meas.	Recall	Prec.	F meas.	Recall	Prec.	F meas.
Accompaniment dB												
BD	65.0%	65.0%	65.0%	63.4%	76.9%	69.5%	61.7%	56.0%	58.7%	**70.0%**	**79.8%**	**74.6%**
SD	55.6%	77.6%	64.8%	61.6%	66.7%	64.1%	51.5%	75.2%	61.1%	**64.2%**	**71.0%**	**67.4%**
HH	81.4%	74.3%	77.7%	80.8%	75.0%	77.8%	82.9%	78.5%	78.7%	**86.5%**	**73.6%**	**79.5%**
Accompaniment 6 dB												
BD	63.3%	70.0%	66.5%	55.8%	63.9%	59.6%	66.5%	75.6%	70.7%	**66.3%**	**78.8%**	**72.1%**
SD	46.7%	70.0%	56.0%	53.4%	66.7%	59.3%	52.0%	66.6%	58.4%	**56.3%**	**68.2%**	**61.7%**
HH	78.2%	69.6%	73.7%	74.2%	72.9%	73.6%	75.7%	74.1%	74.9%	**82.0%**	**71.3%**	**76.3%**
Accompaniment +0 dB												
BD	61.7%	59.5%	60.6%	53.7%	61.5%	57.4%	61.3%	67.0%	64.1%	**65.3%**	**74.4%**	**69.5%**
SD	48.0%	61.7%	54.0%	49.1%	60.8%	54.3%	48.9%	65.7%	56.1%	**55.2%**	**61.9%**	**58.3%**
HH	74.5%	68.2%	71.2%	73.9%	71.4%	72.7%	76.1%	71.6%	73.8%	**81.8%**	**70.2%**	**75.5%**
Accompaniment +6 dB												
BD	61.0%	59.1%	60.0%	53.3%	57.1%	55.1%	61.6%	55.4%	58.3%	**65.0%**	**64.3%**	**64.6%**
SD	39.9%	49.1%	44.0%	46.3%	53.4%	49.6%	44.4%	54.7%	49.0%	**51.6%**	**51.1%**	**51.3%**
HH	69.6%	63.9%	66.6%	61.3%	65.3%	63.2%	67.6%	67.7%	67.7%	**73.5%**	**64.6%**	**68.7%**

strokes of instrument k to be detected. Precision, recall, and F-measure for the instrument k are

$$P = \frac{N_c}{N_d} \quad R = \frac{N_c}{N} \quad \text{F-measure} = \frac{2PR}{P+R}.$$

H. Results

1) Classification Results: Classification results are given in Table I for all the *minus one* sequences of the ENST-drums corpus, and in Table II for its publicly available subset. Results are truncated before the first nonsignificant digit, i.e., the 95% confidence interval has an amplitude smaller than 0.1%.

First, it can be observed that the drum-enhancement only slightly improves (or even degrades, in the case of the bass drum) the result of the classification. The largest performance gains are observed when the accompaniment is louder on the snare drum and hi-hat classes. A more thorough analysis of the classification results reveals that for a fraction of the database, the detection of the bass drum hits from the separated drum signal is less accurate (A difference of up to 7% of the F-measure). For the remaining set, the bass drum detection is more accurate on the drum-enhanced signal. This can be accounted by a difference in the bass drum used between the two sets of sequences. Most sequences are played on a standard rock kit, as commonly used in popular music, whose bass drum produces a very low harmonic component. The only harmonic component in the lowest range of the spectrum is the contribution of the bass drum, which is thus eliminated by the noise subspace projection. Some other sequences are played with a specific Latin drum kit with a smaller bass drum than usual which produces a higher-pitched harmonic component, in the same range as the fundamental frequency of the bass. This component is consequently preserved by the noise subspace projection (the louder harmonic components in this frequency range being those of the bass). The difficulty of generalizing from this specific case to the other ones explains the slightly lower results. This issue can only be avoided with a larger and more diverse (in terms of drum kits) training database.

The fusion algorithms proved to be very successful independently of the accompaniment level. For all instruments, the F-measure scores of the late fusion method is larger than the best scores of the two methods employing only one signal. This

TABLE III
FEATURE SELECTION RESULTS FOR EACH CATEGORY (T=TEMPORAL, E=ENERGY, S=SPECTRAL, C=CEPSTRAL, AND P=PERCEPTUAL)

	Original signal						Drum-enhanced signal					
	T	E	S	C	P	Total	T	E	S	C	P	Total
Accompaniment $-\infty$ dB												
BD	1	5	0	1	1	8	2	0	0	0	0	2
SD	1	1	1	1	1	5	0	2	1	1	1	5
HH	0	2	0	0	1	3	1	1	3	1	1	7
Accompaniment -6 dB												
BD	1	3	0	1	1	6	1	1	0	2	0	4
SD	2	1	0	1	0	4	0	3	0	3	0	6
HH	2	0	0	0	2	4	1	0	3	1	1	6
Accompaniment $+0$ dB												
BD	0	2	0	0	0	2	1	4	0	3	0	8
SD	2	2	0	0	0	4	2	1	0	3	0	6
HH	1	0	0	0	0	1	1	1	5	1	1	9
Accompaniment $+6$ dB												
BD	0	4	0	0	0	4	1	4	0	1	0	6
SD	2	1	0	0	0	3	2	3	0	2	0	7
HH	2	0	0	0	0	2	1	0	4	0	3	8

suggests that the information extracted from the original and the drum-enhanced signal is complementary.

We tested the publicly available system of Tanghe *et al.* [7] on our dataset. Without prior training, it achieved performances similar to those of our system when the drums were predominant, but its performances drastically degraded when the accompaniment music was louder. Since a subset of our database is publicly available (Refer to [35] for more information about its distribution), we encourage other researchers in the field to test their algorithms on this data.

2) Feature Selection Results: To emphasize on the complementarity of features and the validity of the fusion approach, we selected the ten most relevant features among the features extracted from both the original and drum-enhanced signals. The SVM-RFE algorithm was used for this task. The number of selected features in each category was counted. The results are given in Table III

It can be seen that the number of features extracted from the drum-enhanced signal increases with the level of the accompaniment music. The hi-hat and snare drum benefit the most from the features extracted from the drum-enhanced signal. Interestingly, spectral and cepstral features are of little interest when extracted from the original signal. However, they are more frequently selected, when extracted from the drum-enhanced signal. This underlines their lack of robustness to the addition of background accompaniment. On the whole, the most commonly selected features are those related to the energy in typical frequency bands—which are both robust, and specifically designed for the problem of drum transcription. Detailed feature selection results are available online at http://www.tsi.enst.fr/~gillet/pdf/details.pdf.

IV. DRUM TRACK EXTRACTION FROM POLYPHONIC MUSIC

A wide range of methods have been proposed for the separation of audio sources, some of them dedicated to stereo signals (a representative selection of such algorithms are described and evaluated in [36]) some others to monophonic signals. In this case, the separation can be achieved by using a prior model of the sources to be separated (HMMs in [37], Bayesian models in [38], or bags of typical frames in [39]). Other unsupervised methods use psychoacoustic criteria to group related partials [40] or aim to compactly describe the spectrogram as a sum of a few components by means of methods such as ISA [41], NMF [42], or Sparse Coding [43]. Furthermore, several solutions to the specific problem of drum track extraction or resynthesis from polyphonic music have already been proposed (see, for example, [11], [15], and [44]).

In this section, we present several novel approaches that target high-quality remixing applications. First, an extension of our previous method [46] is proposed in Section IV-A. Second, an alternative approach based on time-varying Wiener filtering, along with specific enhancements to the drum separation task, is exposed in Section IV-B. Finally, a comparative evaluation involving state of the art algorithms is provided in Section IV-C.

A. Time/Frequency/Subspace Masking

As seen in Section II-B, a signal can be analyzed in subband harmonic/noise components. Let x_{hk} and x_{rk} be the harmonic and stochastic components of the kth subband signal, respectively. Since a multirate implementation of the filter bank was used, let $\hat{x}_{hk}$ and $\hat{x}_{rk}$ be their full-band versions (after expansion and application of the synthesis filter). Directly reconstructing a signal from the noise components $\sum_{k=1}^{8} \hat{x}_{rk}$ produces a drum-enhanced signal good enough for transcription applications, but whose quality is insufficient for separation and remixing purposes. In order to improve the quality of the reconstruction, we propose to apply different time-varying gains to each of the subband harmonic and stochastic signals: $s = \sum_k \alpha_{hk}\hat{x}_{hk} + \alpha_{rk}\hat{x}_{rk}$. These gains must ensure that only noise and harmonic components associated to drum instruments are present in the reconstruction. For this purpose, we define, for each drum instrument, frequency/subspace temporal envelopes that reflect the distribution of energy in the harmonic and stochastic component of each subband.

1) Extraction of the Frequency/Subspace Temporal Envelopes: The analysis described in Section II-B is performed on a N sample long solo hit of each category i of drum instruments to be considered (bass drum, snare drum, and hi-hat). Let $\hat{x}^i_{hk}$ and $\hat{x}^i_{rk}$ be the resulting harmonic and noise subband signals. The amplitude envelope of each of these signals is fitted with exponentially decaying envelopes, resulting in envelopes e^i_{hk} and e^i_{rk}. This step can be performed on several solo hits for each class of instruments—in which case the corresponding envelopes are averaged.

2) Detection of Drum Events: The next step consists in detecting occurrences of bass drum, snare drum or hi-hat hits from the music signal. Though any transcription method (see Section III) can be used for this task, the frequency/subspace representation and the extracted envelopes can be directly used for this purpose (with suboptimal performances). Actually, a simple drum detection scheme bearing similarity to the

template matching procedure introduced in [12] consists of detecting a hit of the instrument i at the note onset n_0 whenever the quantity $D^i(n_0) > \tau_i$, where τ_i is a threshold and $D^i(n_0)$ is defined as

$$D^i(n_0) = \sum_{k=1}^{8} \sum_{n=0}^{N-1} \left[e^i_{hk}(n)\hat{x}_{hk}(n_0+n) + e^i_{rk}(n)\hat{x}_{rk}(n_0+n) \right]^2 .$$

3) Remasking: Let $\mathbb{I}^i(n)$ be a function equal to 1 if n is the onset of a note played by the drum instrument i, 0 otherwise. The time-varying gains are computed as

$$\alpha_{hk}(n) = \max_i \left(\mathbb{I}^i \star e^i_{hk} \right)(n)$$
$$\alpha_{rk}(n) = \max_i \left(\mathbb{I}^i \star e^i_{rk} \right)(n)$$

where $\star$ denotes convolution. Intuitively, these time-varying gains recreate in each subband and subspace the temporal envelope that the signal would have if it only contained the drum events described by $\mathbb{I}^i(n)$. The use of $\max$ to estimate the spectrum or temporal envelope of a mixture from the spectra and envelopes of individual components has been discussed in [37]. It is also worth noting that the algorithm presented in [46] can be described using the same formalism, with empirically defined binary masks used as e^i_{rk} and $e^i_{hk} = 0$.

B. Separation With Wiener Filtering

1) Overview: In this section, we evaluate and extend a separation technique based on Wiener filtering presented by Benaroya *et al.* in [47], whose principle is briefly recalled here. Considering two stationary Gaussian sources s_1 and s_2 of power spectral density (PSD) $\sigma_1^2(f)$ and $\sigma_2^2(f)$, the optimal estimate of s_i from the mixture $s_1 + s_2$ can be obtained by filtering the mixture with a filter of frequency response $\left(\sigma_i^2(f)/\sigma_1^2(f) + \sigma_2^2(f)\right)$. However, audio sources can only be considered as locally stationary, and cannot be described by a single PSD. To take into account these two phenomena, the sources are assumed to be mixtures of stationary Gaussian processes, with slowly time-varying coefficients: $s_i(n) = \sum_{k \in K_i} a_i(n) b_k(n)$, where $a_i(n) \geq 0$ is slowly varying, $b_k(n)$ is a Gaussian process of PSD σ_k^2, and K_i is a set of indices. The σ_k^2 will further be referred to as spectral templates. In this case, the estimation process consists of the following.[48]:

Step 1) Obtaining a time–frequency representation $\mathcal{S}x(l,m)$ of x by means of the STFT—where l is the frequency bin index, and m a frame index.

Step 2) Decomposing for every time frame m the observed power spectra as a sum of the spectral templates $\mathcal{S}x(l,m) \approx \sum_{k \in K_1 \cup K_2} a_k(m)\sigma_k^2(l)$. A sparsity constraint may be imposed on a_k.

Step 3) Estimating the time–frequency representation of the source s_i as

$$\mathcal{S}s_i(l,m) = \frac{\sum_{k \in K_i} a_k(m)\sigma_k^2(l)}{\sum_{k \in K_1 \cup K_2} a_k(m)\sigma_k^2(l)} \mathcal{S}x(l,m).$$

The decomposition at Step 2 can be performed by a multiplicative update, similar to the one used in NMF algorithms [48].

2) Spectral Templates: This approach requires the estimation of spectral templates for the two sources to be separated. In the case of drum track extraction, a set of spectral templates has to be learnt for the drums, and another set for the background music. It is suggested in [48] to use a clustering algorithm to extract a set of typical PSD from the time–frequency representation of solo signals of each instrument to be separated. In this study, we used $|K_1| = 16$ spectral templates for the drums and $|K_2| = 128$ spectral templates for the background music.

3) Optimization for Drum Separation: We observed that the set of PSD extracted from the drum signals using the correlation-based clustering algorithm presented in [48] contained mixtures, in various proportions, of the snare drum, hi-hat, and bass drum. Such mixtures are redundant, as they can be obtained from more elementary PSD containing solo instruments. We consequently followed another approach, which consisted of extracting the 16 PSD from the training drum signals by NMF. Note that this decomposition is not applied to the background music since it yields too specific spectral components, often reduced to a single frequency peak.

A second improvement is brought about by integrating a simple adaptation procedure. It consists of extending, during the decomposition step, the set of drum spectral templates with the PSD of the stochastic component of x observed for frame m. This choice is motivated by the fact that this additional template is a good estimate of the PSD of the drums and allows in particular to better represent the stochastic part, which is not well taken into account in the main 16 spectral templates.

The third improvement concerns the choice of the window size used to compute the STFT representation. While small windows are efficient for segments containing drum onsets, they imply a low-frequency resolution. Moreover, fast variations of the coefficients $a_k(m)$ between adjacent short windows may produce audible artifacts. Reversely, while longer windows are efficient for segments in which the sustained parts of nonpercussive instruments are predominant, they may induce pre-echo artifacts or smooth the transients in the reconstructed signal. To cope with these limitations, we introduced a window size switching scheme for the time–frequency decomposition. Such schemes are common in audio coders to deal with pre-echo artifacts [49]. Two window sizes are used, $L_1 = 1024$ and $L_2 = 128$. Two dictionaries of spectral templates are learned for these two window sizes. The signal, sampled at 22.05 kHz, is processed by blocks of 1024 samples with a 50% overlap. If the examined block contains a note onset (as detected in Section III-B), it is processed as eight 128-sample long windows, otherwise as a single 1024-sample long window. To ensure a perfect reconstruction, transition windows are applied when switching from one size to the other. Sine windows are used for both the analysis and synthesis steps.

C. Evaluation of Drum Track Separation

The objective evaluation of the drum track separation methods presented here is conducted on the *minus one* sequences included in the public subset of the ENST-drums database (see Section III-G1). The performance metrics used

TABLE IV
SIGNAL-TO-DISTORTION, INTERFERENCE, AND ARTIFACT RATIOS (DECIBELS) FOR THE DRUM SEPARATION ALGORITHMS

	Accompaniment −6 dB			Accompaniment +0 dB			Accompaniment +6 dB		
Method	SDR	SIR	SAR	SDR	SIR	SAR	SDR	SIR	SAR
Variable gain	3.9	11.2	6.1	1.2	5.2	4.9	-3.5	-1.2	3.7
NMF+SVM	5.2	14.4	6.2	2.2	**10.7**	3.5	-1.4	**6.9**	0.2
Spectral modulation	0.7	13.8	1.3	-0.8	8.0	0.9	-3.9	2.1	0.0
Sub-band ICA from stereo signal	5.7	10.0	9.7	0.1	4.9	5.9	-6.3	-2.2	2.6
Noise subspace projection	8.3	10.2	14.5	3.0	4.3	**11.5**	-2.7	-1.6	**8.9**
TFS masking	7.6	14.0	9.6	3.4	6.8	7.7	-2.4	-0.6	6.3
Score-informed TFS masking	7.5	**15.9**	8.7	4.6	10.0	7.1	**0.4**	4.1	4.7
Wiener filter	8.6	10.4	**14.8**	3.1	9.4	5.1	-0.4	4.8	2.9
Wiener filter, enhanced	**10.1**	15.7	12.2	**5.5**	**10.7**	8.0	0.2	5.1	3.9

are those defined in [50]. Let s_d and s_a be, respectively, the original drum and accompaniment signals. The estimate of the drum track $\hat{s}_d$ obtained by the separation methods described above can be projected onto the original drum and accompaniment signals

$$\hat{s}_d = \langle \hat{s}_d, s_d \rangle s_d + \langle \hat{s}_d, s_a \rangle s_a + \epsilon_{\text{artif}}$$

where ϵ_{artif} is the residual of the projection. The signal-to-distortion ratio (SDR) is a global measure of the separation quality, while the signal-to-interference (SIR) and signal-to-artifacts (SAR) ratios, respectively, measure the amount of accompaniment music and separation/reconstruction artifacts remaining in the separated signal. They are defined as follows:

$$\text{SDR} = 10 \log_{10} \frac{\|\langle \hat{s}_d, s_d \rangle s_d\|^2}{\|\langle \hat{s}_d, s_a \rangle s_a + \epsilon_{\text{artif}}\|^2}$$

$$\text{SIR} = 10 \log_{10} \frac{\|\langle \hat{s}_d, s_d \rangle s_d\|^2}{\|\langle \hat{s}_d, s_a \rangle s_a\|^2}$$

$$\text{SAR} = 10 \log_{10} \frac{\|\langle \hat{s}_d, s_d \rangle s_d + \langle \hat{s}_d, s_a \rangle s_a\|^2}{\|\epsilon_{\text{artif}}\|^2}.$$

The results[6] are given in Table IV. *Variable gain* consists of using the drum transcription system presented in Section III to detect the onsets of drum events, and applying a fast decaying exponential envelope with a 100-ms time constant at each drum onset. NMF + SVM is our implementation of the algorithm described in [15]. *Spectral Modulation* is described in [45]. *Sub-band ICA from stereo signal* is the preprocessing for stereo signals detailed in Section II-A, with no further processing. *Noise subspace projection* is the band-wise noise subspace projection used in Section II-B, without the subsequent masking. The four other methods were presented in depth in the previous sections.

For mixtures where the drums are predominant or balanced with the accompaniment, best results are achieved with the modified Wiener filtering method. In all cases, our improvements to this method result in better separation performances. This method also produces good results when the background music is predominant. Comparable results are achieved by the score-informed time/frequency/subspace (TFS) masking. Overall, TFS masking performs better when prior knowledge of the score is available. The improvement brought about by this method over a simple noise subspace projection can be shown by increased SDR and SIR. Nonetheless, noise subspace projection tends to be a "conservative" method in the sense that it introduces fewer artifacts in the extracted signals.

It should also be mentioned that the NMF + SVM system proposed in [15] obtained a high SIR—illustrating the ability of this algorithm to strongly discriminate drum components. However, it obtains, along with spectral modulation, rather low SAR underlining the drawback of methods which reconstruct a drum track from a synthetic time–frequency representation rather than filtering the original signal. They are particularly very sensitive to the problem of phase reconstruction from the STFT. Sound examples for all methods are provided online at http://www.tsi.enst.fr/~gillet/ENST-drums/separation/.

V. DISCUSSION AND FUTURE WORK

Similarly to other audio indexing tasks such as melody detection or musical instrument recognition, drum transcription aims to extract high level information related to a single part of a polyphonic signal. Should it be solved by a prior source separation step to isolate the desired part, or should the signal be globally processed? We argue that both approaches should be followed in parallel, and that in spite of the availability of efficient source separation algorithms, a global approach is still relevant. There is so far no way to model the artifacts introduced by source separation algorithms. As a consequence, the robustness of well-known audio features, when extracted on the output of an elaborate processing—such as source separation—remains unknown. Likewise, while the perceptual interpretation and validity of these features in the case of single instrument signals is well understood, their meaning in the polyphonic case is less obvious. In our work, information fusion and feature selection proved to be an efficient way to compensate for this lack of knowledge. There is nevertheless a need for an in-depth evaluation of the robustness of common audio features to the degradations typically produced by source separation algorithms and to the addition of other music parts.

Our experiments show that obtaining the transcription is easier when the isolated signal is available, and vice versa. This situation bears similarity with estimation problems with hidden variables, in which the set of parameters to estimate (in our case, the drum transcription) and the set of latent variables (in this case, a separated signal, or a model of each drum instrument used in the music piece) are difficult to optimize jointly, but easy with respect to each other. This justifies approaches like [12], and also opens the path for future iterative schemes

[6]Note that the original signals s_d and s_a (before mixing) of the ENST-drums database were also used for the methods that require a training step. Even if this may favor the model-based methods, we believe that the coarseness of the model built and the size of the database should considerably limit this bias. Complementary experiments using the nested cross-validation protocol are under way.

where the transcription and separation steps will be performed in sequence (using a source separation process informed by the score obtained in the previous step) and will then be iterated until convergence. Concurrently, there is an interest to investigate efficient ways to jointly estimate the source and the transcription. This is the philosophy followed by NMF or ISA-based methods, in which the spectral and temporal profiles play the role of simpler intermediate representations for which the joint optimization is easy. However, further processing is needed to accurately recover the source and the transcription from this representation. An interesting direction to follow would be to devise a higher level intermediate representation, closer to the source and the transcription, for which a joint optimization procedure could still be found—an example of such representations being NMF-2D [51].

As for source separation, our results showed that methods which aim to filter or modulate the original signal outperformed those requiring a resynthesis step. Thus, a possible improvement for the NMF-based method [15] could be to use the temporal and spectral profiles to build a time-varying filter applied to the original signal, or equivalently to use the reconstructed spectrogram as a mask applied to the original spectrogram as described in [51]. The two methods that gave the best separation results—TFS masking and Wiener filtering rely on a training step to estimate the spectral templates of the sources to separate. This justifies their good performances, but is also a drawback as it makes them sensitive to the generality of the training set used. For some applications (e.g., a drum level control included in a music player) the separation will be expected to work on a very large range of drum signals, including electronic drums. Interesting directions for further improvements of the Wiener-based approach include the use of more sophisticated adaptation schemes (such as the one proposed in [52] for singing voice separation), perceptually motivated time/frequency representations, or a differentiated processing of the harmonic and stochastic components.

Finally, our work highlighted some inadequacies in the performance measures that should be addressed. Especially, drum source separation is very sensitive to the ability of the separation method to restore and preserve the characteristics of the transients in the original signal. It would thus be very relevant to compare how each method performs on the steady and transient segments of the original signal. Meanwhile, subjective listening tests should be conducted to evaluate the separation quality for real-world applications, such as drum track remixing.

VI. Conclusion

The problems of drum track transcription and separation from polyphonic music signals have been addressed in this article. A complete and accurate drum transcription system integrating a large set of features, optimally selected by feature selection approaches has been built. One of the essential specificities of this novel system relies on the combined use of classification and source separation principles. It is in fact shown that improved performances are attained by fusing the transcription results obtained on the original music signal and on a drum-enhanced version estimated by source separation. The complementarity of the information contained in the original and drum-enhanced signal has been further highlighted by analyzing the results of the feature selection process.

Novel approaches for drum track extraction from polyphonic music were also introduced. The results obtained are very encouraging and already allow very high quality remixing capabilities, especially to modify the drum track level by ± 3 dB. It is worth noting that all proposed algorithms are of relatively low complexity and can run in near real time on standard personal computers.

The approaches proposed also open the path for a number of future incremental improvements including the use of model adaptation for both transcription and source separation, or an iterative analysis scheme that would iteratively transcribe and separate until convergence.

References

[1] S. Dixon, F. Gouyon, and G. Widmer, "Towards characterisation of music via rhythmic patterns," in *Proc. Int. Conf. Music Inf. Retrieval*, 2004, pp. 509–516.

[2] A. Kapur, M. Benning, and G. Tzanetakis, "Query by beatboxing: Music information retrieval for the DJ," in *Proc. Int. Conf. Music Inf. Retrieval*, Oct. 2004, pp. 170–177.

[3] O. Gillet and G. Richard, "Drum loops retrieval from spoken queries," *J. Intell. Inf. Syst.*, vol. 24, no. 2, pp. 159–177, 2005.

[4] D. FitzGerald and J. Paulus, "Unpitched percussion transcription," in *Signal Processing Methods for the Automatic Transcription of Music*, A. Klapuri and M. Davy, Eds. New York: Springer, 2006, pp. 131–162.

[5] F. Gouyon and P. Herrera, "Exploration of techniques for automatic labeling of audio drum tracks," in *Proc. MOSART Workshop Current Directions Comput. Music*, 2001, CD-ROM.

[6] O. Gillet and G. Richard, "Automatic transcription of drum loops," in *Proc. 2004 IEEE Conf. Acoust., Speech, Signal Process.*, May 2004, pp. IV-269–IV-272.

[7] K. Tanghe, S. Degroeve, and B. D. Baets, "An algorithm for detecting and labeling drum events in polyphonic music," in *Proc. 2005 MIREX Evaluation Campaign*, 2005, CD-ROM.

[8] V. Sandvold, F. Gouyon, and P. Herrera, "Percussion classification in polyphonic audio recordings using localized sound models," in *Proc. Int. Conf. Music Inf. Retrieval*, Oct. 2004, pp. 537–540.

[9] O. Gillet and G. Richard, "Drum track transcription of polyphonic music using noise subspace projection," in *Proc. Int. Conf. Music Inf. Retrieval*, Sep. 2005, pp. 92–99.

[10] J. Paulus, "Acoustic modelling of drum sounds with hidden Markov models for music transcription," in *Proc. 2006 IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2006, pp. V-241–V-244.

[11] A. Zils, F. Pachet, O. Delerue, and F. Gouyon, "Automatic extraction of drum tracks from polyphonic music signals," in *Proc. Int. Conf. Web Delivering of Music (WEDELMUSIC2002)*, Dec. 2002, pp. 179–183.

[12] K. Yoshii, M. Goto, and H. G. Okuno, "Automatic drum sound description for real-world music using template adaptation and matching methods," in *Proc. Int. Conf. Music Inf. Retrieval*, Oct. 2004, pp. 184–191.

[13] D. FitzGerald, B. Lawlor, and E. Coyle, "Prior subspace analysis for drum transcription," in *Proc. 11AES Conv.*, Mar. 2003, CD-ROM.

[14] C. Uhle and C. Dittmar, "Further steps towards drum transcription of polyphonic music," in *Proc. 11AES Conv.*, May 2004, CD-ROM.

[15] M. Helén and T. Virtanen, "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine," in *Proc. Eur. Signal Process. Conf.*, 2005, CD-ROM.

[16] M. Alonso, "Extraction of metrical information from acoustic music signals," Ph.D. dissertation, ENST, Paris, France, 2006.

[17] D. Barry, B. Lawlor, and E. Coyle, "Sound source separation: Azimuth discrimination and resynthesis," in *Proc. Int. Conf. Digital Audio Effects*, Oct. 2004, CD-ROM.

[18] R. Badeau, R. Boyer, and B. David, "EDS parametric modeling and tracking of audio signals," in *Proc. Int. Conf. Digital Audio Effects*, Sep. 2002, pp. 139–144.

[19] R. Badeau, B. David, and G. Richard, "Selecting the modeling order for the ESPRIT high resolution method: An alternative approach," in *Proc. 2004 Int. Conf. Acoust., Speech, Signal Process.*, May 2005, pp. II-1025–II-1028.

[20] M. Alonso, G. Richard, and B. David, "Extracting note onsets from musical recordings," in *Proc. 2005 IEEE Int. Conf. Multimedia Expo*, 2005, pp. 1–4.
[21] F. Gouyon, P. Herrera, and A. Dehamel, "Automatic labeling of unpitched percussion sounds," in *Proc. 11AES Conv.*, Mar. 2003, CD-ROM.
[22] S. Degroeve, K. Tanghe, B. D. Baets, M. Leman, and J. P. Martens, "A simulated annealing optimization of audio features for drum classification," in *Proc. Int. Conf. Music Inf. Retrieval*, 2005, pp. 482–487.
[23] S. Essid, G. Richard, and B. David, "Musical instrument recognition by pairwise classification strategies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1401–1412, Jul. 2006.
[24] G. Peeters, "A large set of audio features for sound description (Similarity and Classification) in the cuidado project," IRCAM, 2004.
[25] I. Guyon and A. Elisseeff, "An introduction to feature and variable selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.
[26] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, no. 1-2, pp. 273–324, 1997.
[27] R. Fiebrink and I. Fujinaga, "Feature selection pitfalls and music classification," in *Proc. Int. Conf. Music Inf. Retrieval*, 2006, pp. 340–341.
[28] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 301–312, Mar. 2002.
[29] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, no. 1–3, pp. 389–422, 2002.
[30] G. Peeters, "Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization," in *Proc. 11AES Conv.*, Oct. 2003, CD-ROM.
[31] B. Schölkopf and A. J. Smola, *Learning With Kernels*. Cambridge, MA: MIT Press, 2002.
[32] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.
[33] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," in *Advances in Large Margin Classifiers*. Cambridge, MA: MIT Press, 2000, pp. 61–74.
[34] I. Bloch, "Information combination operators for data fusion: A comparative review with classification," in *Proc. SPIE/EUROPTO Conf. Image Signal Process. for Remote Sensing*, Rome, Italy, Sep. 1994, vol. 2315, pp. 148–159.
[35] O. Gillet and G. Richard, "Enst-drums: An extensive audio-visual database for drum signals processing," in *Proc. Int. Conf. Music Inf. Retrieval*, 2006, pp. 156–159.
[36] E. Vincent, H. Sawada, P. Bofill, S. Makino, and J. P. Rosca, "First stereo audio source separation evaluation campaign: Data, algorithms and results," in *Proc. Int. Conf. Independent Compon. Anal. Signal Separation (ICA'07)*, 2007, CD-ROM.
[37] S. T. Roweis, "One microphone source separation," in *Advances in Neural Information Processing Systems*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. Cambridge, MA: MIT Press, 2001, vol. 13, pp. 793–799.
[38] E. Vincent and X. Rodet, "Underdetermined source separation with structured source priors," in *Proc. Symp. Independent Compon. Anal. Blind Signal Separation (ICA'04)*, Apr. 2004, CD-ROM.
[39] D. Ellis and R. Weiss, "Model-based monaural source separation using a vector-quantized phase-vocoder representation," in *Proc. 2006 IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2006, pp. V-957–V-960.
[40] D. Ellis, "Prediction-Driven Computational Auditory Scene Analysis," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, MA, 1996.
[41] M. Casey and A. Westner, "Separation of mixed audio sources by independent subspace analysis," in *Proc. Int. Comput. Music Conf.*, 2000, pp. 154–161.
[42] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Adv. Neural Inf. Process. Syst.*, 2001, vol. 13, pp. 556–562, CD-ROM.
[43] T. Virtanen, "Sound source separation using sparse coding with temporal continuity objective," in *Proc. 2003 Int. Comput. Music Conf.*, 2003, pp. 231–234.
[44] K. M. G. Yoshii and H. G. Okuno, "Inter:D: A drum sound equalizer for controlling volume and timbre of drums," in *Proc. Eur. Workshop Integration of Knowledge, Semantics, Digital Media Technol.*, 2005, CD-ROM.
[45] D. Barry, D. FitzGerald, E. Coyle, and B. Lawlor, "Drum source separation using percussive feature detection and spectral modulation," in *Proc. Irish Signals Syst. Conf. (ISSC'05)*, 2005, CD-ROM.
[46] O. Gillet and G. Richard, "Extraction and remixing of drum tracks from polyphonic music signals," in *Proc. 2005 IEEE Workshop Applicat. Signal Process. to Audio Acoust.*, Oct. 2005, pp. 315–318.
[47] L. Benaroya, F. Bimbot, and R. Gribonval, "Audio source separation with a single sensor," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 191–199, Jan. 2006.
[48] L. Benaroya, L. M. Donagh, F. Bimbot, and R. Gribonval, "Non-negative sparse representation for Wiener based source separation with a single sensor," in *Proc. 2003 IEEE Conf. Acoust., Speech, Signal Process.*, 2003, pp. VI-613–VI-616.
[49] M. Bosi and E. Goldberg, *Introduction to Digital Audio Coding and Standards*. Norwell, MA: Kluwer, 2002.
[50] R. Gribonval, L. Benaroya, E. Vincent, and C. Févotte, "Proposals for performance measurement in source separation," in *Proc. Conf. Ind. Compon. Anal. Blind Signal Separation*, Apr. 2003.
[51] M. N. Schmidt and M. Mørup, "Nonnegative matrix factor 2-d deconvolution for blind single chanel source separation," in *Proc. Symposium on Independent Component Analysis and Blind Signal Separation (ICA'2006)*, 2006, CD-ROM.
[52] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot, "One microphone singing voice separation using source-adapted models," in *Proc. 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Mohonk, NY, 2005, pp. 90–93.

Olivier Gillet (A'07) received the State Engineering degree from the École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 2003, the M.Sc. (DEA) degree in artificial intelligence and pattern recognition from the Université Pierre et Marie Curie (Paris 6) in 2003, and the Ph.D. degree from ENST in 2007, after completing a thesis on drum signal processing and music video analysis.

He joined Google, Zurich, Switzerland, in October 2007 as a software engineer. His interests include signal processing and machine learning for audio content analysis and the integration of video information into music information retrieval systems.

Gaël Richard (SM'06) received the State Engineering degree from the École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1990, the Ph.D. degree from LIMSI-CNRS, University of Paris-XI, in 1994 in speech synthesis and the *Habilitation a Diriger des Recherches* degree from the University of Paris XI in September 2001.

After the Ph.D. degree, he spent two years at the CAIP Center, Rutgers University, Piscataway, NJ, in the Speech Processing Group of Prof. J. Flanagan, where he explored innovative approaches for speech production. From 1997 and 2001, he successively worked for Matra Nortel Communications, Bois d'Arcy, France, and for Philips Consumer Comunications, Montrouge, France. In particular, he was the Project Manager of several large-scale European projects in the field of audio and multimodal signal processing. In September 2001, he joined the Department of Signal and Image Processing, GET-Télécom Paris (ENST), where he is now a Full Professor in audio signal processing and Head of the Audio, Acoustics and Waves Research Group. He is coauthor of over 70 papers and inventor in a number of patents. He is also one of the experts of the European Commission in the field of audio signal processing and man/machine interfaces.

Prof. Richard is a member of EURASIP and is an Associate Editor of the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING.