Signal Processing 91 (2011) 2836–2851

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Recursive nearest neighbor search in a sparse and multiscale domain for comparing audio signals

Bob L. Sturm [a,*,♣], Laurent Daudet [b,♣♣]

[a] Department of Architecture, Design and Media Technology, Aalborg University Copenhagen, Lautrupvang 15, 2750 Ballerup, Denmark
[b] Institut Langevin (LOA), Université Paris Diderot – Paris 7, UMR 7587, 10, rue Vauquelin, 75231 Paris, France

ARTICLE INFO

Article history:
Received 5 January 2010
Received in revised form
15 February 2011
Accepted 2 March 2011
Available online 10 March 2011

Keywords:
Multiscale decomposition
Sparse approximation
Time—frequency dictionary
Audio similarity

ABSTRACT

We investigate recursive nearest neighbor search in a sparse domain at the scale of audio signals. Essentially, to approximate the cosine distance between the signals we make pairwise comparisons between the elements of localized sparse models built from large and redundant multiscale dictionaries of time–frequency atoms. Theoretically, error bounds on these approximations provide efficient means for quickly reducing the search space to the nearest neighborhood of a given data; but we demonstrate here that the best bound defined thus far involving a probabilistic assumption does not provide a practical approach for comparing audio signals with respect to this distance measure. Our experiments show, however, that regardless of these non-discriminative bounds, we only need to make a few atom pair comparisons to reveal, e.g., the origin of an excerpted signal, or melodies with similar time–frequency structures.

1. Introduction

Sparse approximation is essentially the modeling of data with few terms from a large and typically overcomplete set of atoms, called a "dictionary" [24]. Consider an $\mathbf{x} \in \mathbb{R}^K$, and a dictionary $\mathcal{D}$ composed of N unit-norm atoms in the same space, expressed in matrix form as $\mathbf{D} \in \mathbb{R}^{K \times N}$, where $N \gg K$. A pursuit is an algorithm that decomposes $\mathbf{x}$ in terms of $\mathbf{D}$ such that $\|\mathbf{x}-\mathbf{D}\mathbf{s}\|^2 \leq \varepsilon$ for some error $\varepsilon \geq 0$. (In this paper, we work in a Hilbert space unless otherwise noted.) When $\mathbf{D}$ is overcomplete, $\mathbf{D}$ has full row rank and there exists an infinite number of solutions to choose from, even for $\varepsilon = 0$. Sparse approximation aims to find a solution $\mathbf{s}$ that is mostly zeros for ε small. In that case, we say that $\mathbf{x}$ is sparse in $\mathcal{D}$.

Matching pursuit (MP) is an iterative descent sparse approximation method based on greedy atom selection [17,24]. We express the nth-order model of the signal $\mathbf{x} = \mathbf{H}(n)\mathbf{a}(n) + \mathbf{r}(n)$, where $\mathbf{a}(n)$ is a length-n vector of weights, $\mathbf{H}(n)$ are the n corresponding columns of $\mathbf{D}$, and $\mathbf{r}(n)$ is the residual. MP augments the nth-order representation, $\mathcal{X}_n = \{\mathbf{H}(n), \mathbf{a}(n), \mathbf{r}(n)\}$, according to

$$\mathcal{X}_{n+1} = \left\{ \begin{array}{l} \mathbf{H}(n+1) = [\mathbf{H}(n) | \mathbf{h}_n] \\ \mathbf{a}(n+1) = [\mathbf{a}^T(n), \langle \mathbf{r}(n), \mathbf{h}_n \rangle]^T \\ \mathbf{r}(n+1) = \mathbf{x} - \mathbf{H}(n+1)\mathbf{a}(n+1) \end{array} \right\} \quad (1)$$

using the atom selection criterion

$$\mathbf{h}_n = \arg \min_{\mathbf{d} \in \mathcal{D}} \|\mathbf{r}(n) - \langle \mathbf{r}(n), \mathbf{d} \rangle \mathbf{d}\|^2 = \arg \max_{\mathbf{d} \in \mathcal{D}} |\langle \mathbf{r}(n), \mathbf{d} \rangle| \quad (2)$$

where $\|\mathbf{d}\| = 1$ is implicit. The inner product here is defined $\langle \mathbf{x}, \mathbf{y} \rangle \triangleq \mathbf{y}^T\mathbf{x}$. This criterion guarantees $\|\mathbf{r}(n+1)\|^2 \leq \|\mathbf{r}(n)\|^2$ [24]. Other sparse approximation methods include orthogonal MP [28], orthogonal least squares (OLS) [41,33], molecular methods [9,38,19], cyclic MP and OLS [36], and minimizing error jointly with a relaxed sparsity measure [6]. These approaches have higher computational

* Corresponding author.
E-mail addresses: boblsturm@gmail.com (B.L. Sturm), laurent.daudet@espci.fr (L. Daudet).
♣ EURASIP# 7255.
♣♣ EURASIP # 2298.

0165-1684/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2011.03.002

complexities than MP, but can produce data models that are more sparse.

Sparse approximation is data-adaptive and can produce parametric and multiscale models having features that function more like mid-level "objects" than low-level projections onto sets of vectors [9,19,38,8,22,27,32,34,43]. These aspects make sparse approximation a compelling complement to state-of-the-art approaches for and applications of comparing audio signals based upon, e.g., monoresolution cepstral and redundant time–frequency representations, such as fingerprinting [42], cover song identification [26,10,3,35], content segmentation, indexing, search and retrieval [16,4], artist or genre classification [39].

In the literature we find some existing approaches to working with audio signals in a sparse domain. Features built from sparse approximations can provide competitive descriptors for music information retrieval tasks, such as beat tracking, chord recognition, and genre classification [40,32]. Sparse representation classifiers have been applied to music genre recognition [27,5], and robust speech recognition [12]. Parameters of sparse models can be compared using histograms to find similar sounds in acoustic environment recordings [7,8], or atoms can be learned to compare and group percussion sounds [34]. Biologically inspired sparse codings of correlograms of sounds can be used to learn associations between descriptive high-level keywords and audio features such that new sounds can be automatically categorized, and large collections of sounds can be queried in meaningful ways [22]. Outside the realm of audio signals, sparse approximation has been applied to face recognition [44], object recognition [29], and landmine detection [25].

In this paper, we discuss the comparison of audio signals in a sparse domain, but not specifically for fingerprinting or efficient audio indexing and search—two tasks that have been convincingly solved [42,13,16,18]. We explore the possibilities and effectiveness of comparing, atom-by-atom, audio signals modeled using sparse approximation and large overcomplete time–frequency dictionaries. Our contributions are threefold: (1) we generalize recursive nearest-neighbor search algorithm to comparing subsequences [14,15]; (2) we show that though sparse models of audio signals can be compared by considering pairs of atoms, the best bound so far derived [14,15] does not make a practical procedure; and (3) we show experimentally that the hierarchic comparison of audio signals in a sparse domain still provides intriguing and informative results. Overall, our work here shows that a sparse domain can facilitate comparisons of audio signals in "hierarchical" ways through comparing individual elements of each sparse data model organized roughly in order of importance.

In the next two sections, we discuss and elaborate upon a recursive method of nearest neighbor search in a sparse domain [14,15]. We extend this method to comparing subsequences, and examine the practicality of probabilistic bounds on the distances between neighbors. In the fourth section, we describe several experiments in which we compare a variety of audio signals through their sparse models. We conclude with a discussion about the results and several future directions.

2. Nearest neighbor search by recursion in a sparse domain

Consider a set of signals

$$\mathcal{Y} \triangleq \{\mathbf{y}_i \in \mathbb{R}^K : \|\mathbf{y}_i\| = 1\}_{i\in\mathcal{I}} \tag{3}$$

where $\mathcal{I} = \{1,2,\ldots\}$ indexes this set, and a query signal $\mathbf{x}_q \in \mathbb{R}^K$, $\|\mathbf{x}_q\| = 1$. Assume that we have generated sparse approximations for all of these signals $\hat{\mathcal{Y}} \triangleq \{\{\mathbf{H}_i(n_i), \mathbf{a}_i(n_i), \mathbf{r}_i(n_i)\} : \mathbf{y}_i = \mathbf{H}_i(n_i)\mathbf{a}_i(n_i) + \mathbf{r}_i(n_i)\}_{i\in\mathcal{I}}$ using a dictionary $\mathbf{D}$ that spans the space $\mathbb{R}^K$, and giving the n_q-order representation $\{\mathbf{H}_q(n_q), \mathbf{a}_q(n_q), \mathbf{r}_q(n_q)\}$ for $\mathbf{x}_q$. Since $\mathcal{D}$ spans $\mathbb{R}^K$, $\mathcal{D}$ is "complete," and any signal in $\mathbb{R}^K$ is "compressible" in $\mathcal{D}$, meaning that we can order the representation weights in $\mathbf{a}_i(n_i)$ or $\mathbf{a}_q(n_q)$ in terms of decreasing magnitude, i.e.,

$$0 < |[\mathbf{a}_i(n_i)]_{m+1}| \le |[\mathbf{a}_i(n_i)]_m| \le Cm^{-\gamma}, \quad m = 1,2,\ldots,n_i - 1 \tag{4}$$

for n_i arbitrarily large, with $C > 0$, and where $[\mathbf{a}]_m$ is the mth element of the column vector $\mathbf{a}$. This can be seen in the magnitude representation weights in Fig. 1, which are weights of sparse representations of piano notes, described in Section 4.1. With MP and a complete dictionary, we are guaranteed $\gamma > 0$ because $\|\mathbf{r}(n+1)\|^2 < \|\mathbf{r}(n)\|^2$ for all n [24].

Consider the Euclidean distance between two signals of the same dimension, which is the cosine distance for unit-norm signals. Thus, with respect to this distance, the $\mathbf{y}_i \in \mathcal{Y}$ nearest to $\mathbf{x}_q$ is given by solving

$$\min_{i\in\mathcal{I}} \|\mathbf{y}_i - \mathbf{x}_q\| = \max_{i\in\mathcal{I}} \langle \mathbf{x}_q, \mathbf{y}_i \rangle \tag{5}$$

We can express this inner product in terms of sparse representations

$$\begin{aligned}\langle \mathbf{x}_q, \mathbf{y}_i \rangle &= \langle \mathbf{H}_q(n_q)\mathbf{a}_q(n_q) + \mathbf{r}_q(n_q), \mathbf{H}_i(n_i)\mathbf{a}_i(n_i) + \mathbf{r}_i(n_i) \rangle \\ &= \mathbf{a}_i^T(n_i)\mathbf{H}_i^T(n_i)\mathbf{H}_q(n_q)\mathbf{a}_q(n_q) + O[\mathbf{r}_q, \mathbf{r}_i]\end{aligned} \tag{6}$$

With a complete dictionary we can make $O[\mathbf{r}_q, \mathbf{r}_i]$ negligible by choosing ε arbitrarily small, so we can

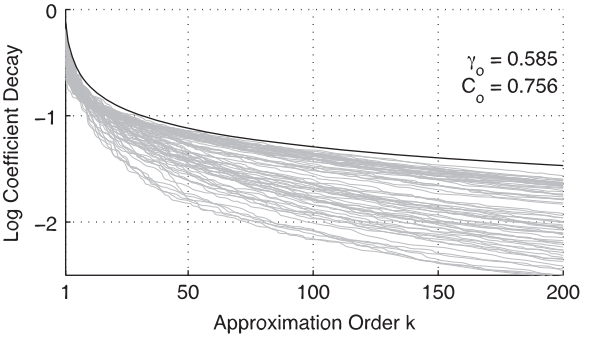

Fig. 1. Gray lines show decays of representation weight magnitudes as a function of approximation order k for several decompositions of unit-norm signals (4-s recordings of single piano notes described in Section 4.1). Thick black line shows a global compressibility bound with its parameters.

express (5) as

$$\max_{i\in\mathcal{I}} \langle \mathbf{x}_q,\mathbf{y}_i\rangle \sim \max_{i\in\mathcal{I}} \mathbf{a}_i^T(n_i)\mathbf{G}_{iq}\mathbf{a}_q(n_q) = \max_{i\in\mathcal{I}} \sum_{m=1}^{n_i}\sum_{l=1}^{n_q}[\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{ml} \tag{7}$$

where $[\mathbf{B}\bullet\mathbf{C}]_{ml}=[\mathbf{B}]_{ml}[\mathbf{C}]_{ml}$ is the Hadamard, or entry wise, product, $[\mathbf{B}]_{ml}$ is the element of $\mathbf{B}$ in the mth row of the lth column, $\mathbf{G}_{iq}\triangleq\mathbf{H}_i^T(n_i)\mathbf{H}_q(n_q)$ is a $n_i\times n_q$ matrix with elements from the Gramian of the dictionary, i.e., $\mathbf{G}\triangleq\mathbf{D}^T\mathbf{D}$, and finally we define the outer product of the weights

$$\mathbf{A}_{iq}\triangleq\mathbf{a}_i(n_i)\mathbf{a}_q^T(n_q). \tag{8}$$

2.1. Recursive search limited by bounds

Since we expect the decay of the magnitude of elements in $\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}$ to be fastest in diagonal directions by (4), we define a recursive sum along the M anti-diagonals starting at the top left:

$$S_{iq}(M)\triangleq S_{iq}(M-1)+\sum_{m=1}^{M}[\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{m(M-m+1)} \tag{9}$$

for $M=2,3,\ldots,\min(n_i,n_q)$, and setting $S_{iq}(1)=[\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{11}$. With this we can express the argument of (7) as

$$\langle \mathbf{x}_q,\mathbf{y}_i\rangle \approx \sum_{m=1}^{n_i}\sum_{l=1}^{n_q}[\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{ml} = S_{iq}(M)+R(M) \tag{10}$$

where at step M, we are comparing M additional pairs of atoms to those considered in the previous steps. $R(M)$ is a remainder that we will bound. The total number of atom pairs contributing to $S_{iq}(M)$ (9) is

$$P(M)\triangleq\sum_{m=1}^{M} m = M(M+1)/2. \tag{11}$$

The approach taken by Jost et al. [14,15] to find the nearest neighbors of $\mathbf{x}_q$ in $\mathcal{Y}$ bounds the remainder $R(M)$ by compressibility (4). Assuming we have a positive upper bound on the remainder, i.e., $R(M)\le\tilde{R}(M)$, we know lower and upper bounds on the cosine distance $L_{iq}(M)\le\langle \mathbf{x}_q,\mathbf{y}_i\rangle\le U_{iq}(M)$, where

$$L_{iq}(M)\triangleq S_{iq}(M)-\tilde{R}(M) \tag{12}$$

$$U_{iq}(M)\triangleq S_{iq}(M)+\tilde{R}(M) \tag{13}$$

Finding elements of $\mathcal{Y}$ close to $\mathbf{x}_q$ with respect to (5) can be done recursively over the approximation order M. For a given M, we find $\{S_{iq}(M)\}_{i\in\mathcal{I}}$, compute the remainder $\tilde{R}(M)$, and eliminate signals that are not sufficiently close to $\mathbf{x}_q$ with respect to their cosine distance by comparing the bounds. This approach is similar to hierarchical ones, e.g., [21], where the features become more discriminable as the search process runs. (Also note that compressibility is similar to the argument made in justifying the truncation of Fourier series in early work on similarity search [1,11,30], i.e., that power spectral densities of many time-series decay like $\mathcal{O}(|f|^{-b})$ with $b>1$.)

Starting with $M=1$, we compute the sets $\{L_{iq}(1)\}_{i\in\mathcal{I}}$ and $\{U_{iq}(1)\}_{i\in\mathcal{I}}$, that is, the first-order upper and lower bounds of the set of distances of $\mathbf{x}_q$ from all signals in $\mathcal{Y}$. Then we find the index of the largest lower bound $i_{\max}=\arg\max_{i\in\mathcal{I}}L_{iq}(1)$, and reduce the search space to $\mathcal{I}_1\triangleq\{i\in\mathcal{I}:U_{iq}(1)\ge L_{i_{\max}q}(1)\}$, since all other data have a least upper bound on their inner product with $\mathbf{x}_q$ than the greatest lower bound in the set. For the next step, we compute the sets $\{L_{iq}(2)\}_{i\in\mathcal{I}_1}$ and $\{U_{iq}(2)\}_{i\in\mathcal{I}_1}$, find the index of the maximum $i_{\max}=\arg\max_{i\in\mathcal{I}_1}L_{iq}(2)$, and construct the reduced set $\mathcal{I}_2\triangleq\{i\in\mathcal{I}_1:U_{iq}(2)\ge L_{i_{\max}q}(2)\}$. Continuing in this way, we find the elements of $\mathcal{Y}$ closest to $\mathbf{x}_q$ at each M with respect to the cosine distance by recursing into the sparse approximations of the signals.

2.2. Bounding the remainder

To reduce the search space quickly we desire that (12) and (13) converge quickly to the neighborhood of $\langle \mathbf{x}_q,\mathbf{y}_i\rangle$, or in other words, that the bounds on the remainder quickly become discriminative. Jost et al. [14,15] derive three different bounds on $R(M)$. From the weakest to the strongest, these are:

1. $[\mathbf{G}_{iq}]_{ml}=1$ (worst case scenario, but impossible for $n>1$)
$$R(M)\le C^2(\|\mathbf{c}_M^\gamma\|_1+\|\mathbf{d}^\gamma\|_1) \tag{14}$$
2. $[\mathbf{G}_{iq}]_{ml}\sim$ iid Bernoulli(0.5), $\Omega=\{-1,1\}$ (impossible for $n>1$)
$$R(M)\le C^2\sqrt{\ln 4}(\|\mathbf{c}_M^\gamma\|_2^2+\|\mathbf{d}^\gamma\|_2^2)^{1/2} \tag{15}$$
3. $[\mathbf{G}_{iq}]_{ml}\sim$ iid Uniform, $\Omega=[-1,1]$,
$$R(M)\le C^2\sqrt{2/3}\,\mathrm{Erf}^{-1}(p)(\|\mathbf{c}_M^\gamma\|_2^2+\|\mathbf{d}^\gamma\|_2^2)^{1/2} \tag{16}$$
with probability $0\le p\le 1$

where we define the following vectors for $n\triangleq\min(n_i,n_q)$ and $M=2,\ldots,n$:

$$\mathbf{c}_M^\gamma\triangleq\{[l(m-l+1)]^{-\gamma}:m=M+1,\ldots,n;\ l=1,\ldots,m\} \tag{17}$$

$$\mathbf{d}^\gamma\triangleq\{[l(n-m+1)]^{-\gamma}:m=1,\ldots,n-1;\ l=m+1,\ldots,n\}. \tag{18}$$

Appendix A gives derivations of these bounds, as well as the efficient computation of (16) for the special case of $\gamma=0.5$. The parameters (C,γ) describe the compressibility of the signals in the dictionary (4). The bounds of (15) and (16) are much more discriminative than (14) because they involve an ℓ_2-norm at the price of uncertainty in the bound. The bound in (16) is attractive because we can tune it with the parameter p, which is the probability that the remainder will not exceed the bound. Fig. 2 shows bounds based on (16) for several pairs of compressibility parameters for the dataset used to produce Fig. 1.

2.3. Estimating the compressibility parameters

The bounds (14)–(16), and consequently the number of atom pairs we must consider before the bounds

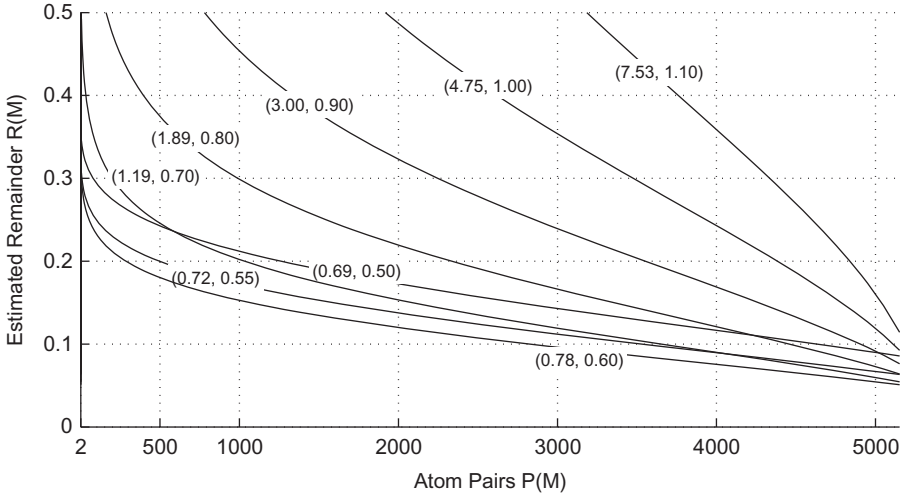

Fig. 2. Estimated remainder (assuming unit-norm signals) using bound in (16) with $p=0.2$ (probability that remainder does not exceed bound) and $n=100$ (number of elements in each sparse model) as a function of the number of atom pairs already considered for several pairs of compressibility parameters (C,γ) estimated from the dataset used to produce Fig. 1.

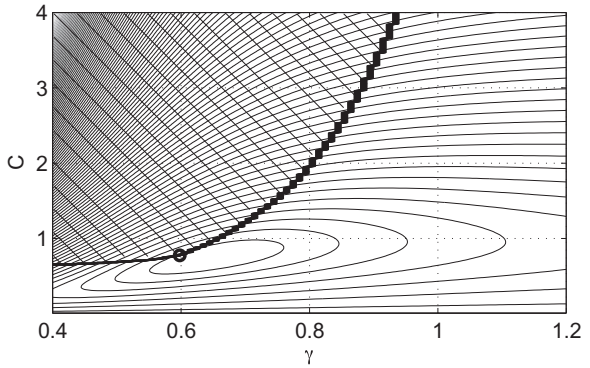

Fig. 3. Error surface as a function of the compressibility parameters for the dataset used to produce Fig. 1, with the feasible set shaded at top left, and optimal parameters marked by a circle.

become discriminable, depend on the compressibility parameters, (C,γ)—which themselves depend in a complex way on the signal, the dictionary, and the method of sparse approximation. Fig. 3 shows the error surface, feasible set, and the optimal parameters for the dataset used to produce Fig. 1. We describe our parameter estimation procedure in Appendix B. The resulting bound is shown in black in Fig. 1. These compressibility parameters also agree with those seen in Fig. 2.

3. Subsequence nearest neighbor search in a localized sparse domain

The recursive nearest neighbor search so far described has the obvious limitation that it cannot be applied to comparing subsequences of large data vectors, as is natural for comparing audio signals. Thus, we must adapt its structure to work for comparing subsequences in a set of data

$$\mathcal{Y} \triangleq \{\mathbf{y}_i \in \mathbb{R}^{N_i} : N_i \geq K\}_{i\in\mathcal{I}} \tag{19}$$

(note that now we do not restrict the norms of these signals). We can create from the elements of $\mathcal{Y}$ a new set of all subsequences having the same length as a K-dimensional query $\mathbf{x}_q$ $(K<N_i)$:

$$\mathcal{Y}_K \triangleq \{\mathbf{P}_t\mathbf{y}_i/\|\mathbf{P}_t\mathbf{y}_i\| : t\in\mathcal{T}_i=\{1,2,\ldots N_i-K+1\},\mathbf{y}_i\in\mathcal{Y}\} \tag{20}$$

where $\mathbf{P}_t$ extracts a K-length subsequence in $\mathbf{y}_i$ starting a time-index t (it is an identity matrix of size K starting a column t in a $K\times N_i$ matrix of zeros). The set $\mathcal{T}_i$ are times at which we create length-K subsequences from $\mathbf{y}_i$. If we decompose each of these by sparse approximation, then we can use the framework in the previous section. However, sparse approximation is an expensive operation that we want to do only once for the entire signal, and independent of the length of $\mathbf{x}_q$.

To address this problem, we instead approximate each element in $\mathcal{Y}_K$ by building local sparse representations from the global sparse approximations of each $\mathbf{y}_i$, and then calculating their distance to $\mathbf{x}_q$ using the framework in the previous section. From here on we consider only the K-length subsequences of a single element $\mathbf{y}_i\in\mathcal{Y}$ without loss of generality (i.e., all other elements of $\mathcal{Y}$ can be included as subsequences). Toward this end, consider that we have decomposed the N_i-length signal $\mathbf{y}_i$ using a complete dictionary to produce the representation $\{\mathbf{H}_i(n_i),\mathbf{a}_i(n_i),\mathbf{r}_i(n_i)\}$. From this we construct the local sparse representations of $\mathbf{y}_i$:

$$\hat{\mathcal{Y}}_K \triangleq \{\{\mathbf{P}_t\mathbf{H}_i(n_i),\xi_t\mathbf{a}_i(n_i),\mathbf{P}_t\mathbf{r}_i(n_i)\} : t\in\mathcal{T}_i\} \tag{21}$$

where the time partition $\mathcal{T}_i$ is the set of all times at which we extract a K-length subsequence from $\mathbf{y}_i$, and ξ_t is set such that $\|\xi_t\mathbf{P}_t\mathbf{y}_i\|=1$, i.e., each length-$K$ subsequence is unit-norm. For each K-dimensional subsequence,

(7) now becomes

$$\max_{t\in\mathcal{T}_i} \langle \mathbf{x}_q, \mathbf{P}_t\mathbf{y}_i\rangle = \max_{t\in\mathcal{T}_i}[\langle \mathbf{H}_q(n_q)\mathbf{a}_q(n_q), \xi_t\mathbf{P}_t\mathbf{H}_i(n_i)\mathbf{a}_i(n_i)\rangle + O[\mathbf{r}_q,\mathbf{r}_i]]$$
$$\approx \max_{t\in\mathcal{T}_i} \xi_t \mathbf{a}_i^T(n_i)\mathbf{H}_i^T(n_i)\mathbf{P}_t^T\mathbf{H}_q(n_q)\mathbf{a}_q(n_q)$$
$$= \max_{t\in\mathcal{T}_i} \xi_t \sum_{m=1}^{n_i}\sum_{l=1}^{n_q}[\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}(t)]_{ml} \quad (22)$$

where $\mathbf{A}_{iq}$ is defined in (8), we define the time-localized Gramian

$$\mathbf{G}_{iq}(t) \triangleq \mathbf{H}_i^T(n_i)\mathbf{P}_t^T\mathbf{H}_q(n_q) \quad (23)$$

and we have excluded the terms involving the residuals because we can make them arbitrarily small.

3.1. Estimating the localized energy

The only thing left to do is to find an expression for ξ_t so that each subsequence is comparable with the others with respect to the cosine distance. We assume that the localized energy can be approximated from the local sparse representation in the following way assuming $\|\mathbf{P}_t\mathbf{y}_i\| > 0$

$$\xi_t = \|\mathbf{P}_t\mathbf{y}_i\| \approx \sqrt{\mathbf{a}_i^T(n_i)\mathbf{H}_i^T(n_i)\mathbf{P}_t^T\mathbf{P}_t\mathbf{H}_i(n_i)\mathbf{a}_i(n_i)} \approx \sqrt{\sum_{j=1}^{n_t} w_j a_j^2} \quad (24)$$

where the n_t weights $a_j \in \{[\mathbf{a}_i(n_i)]_m : [\mathbf{H}_i^T(n_i)\mathbf{P}_t^T\mathbf{P}_t\mathbf{H}_i\,(n_i)]_{ml} \neq 0, 1\le m,l\le n_i\}$ are those associated with atoms having support in $[t,t+K)$, and w_j we define to weigh the contribution of a_j^2 to the localized energy estimate. We set $\xi_t = 0$ if $\sum_{j=1}^{n_t} a_j^2 = 0$.

If all atoms contributing to the subsequence have their entire support in $[t,t+K)$, and are orthonormal, then we can set each $w_j = 1$. This does not hold for subsequences of a signal decomposed using an overcomplete dictionary, as shown by Fig. 4. For much of the time we see $\sum_{j=1}^{n_t} a_j^2 \ge \|\mathbf{P}_t\mathbf{y}_i\|^2$, which means our localized estimate of the segment energy is greater than its real value. This will make ξ_t and consequently (22) smaller.

Instead, we make a more reasonable estimate of $\|\mathbf{P}_t\mathbf{y}_i\|$ by accounting for the fact that atoms can have support outside $[t,t+K)$. For instance, if an atom has some fraction of support in the subsequence we multiply its weight by that fraction. We thus weigh the contribution of the jth atom to the subsequence norm using

$$w_j = \begin{cases} 1, & u_j \ge t, u_j+s_j \le t+K \\ (K/s_j)^2, & u_j < t, u_j+s_j \ge t+K \\ (u_j+s_j-t)^2/s_j^2, & u_j < t, t < u_j+s_j \le t+K \\ (t+K-u_j)^2/s_j^2, & t \le u_j < t+K, u_j+s_j > t+K \end{cases} \quad (25)$$

where u_j and s_j are the position and scale, respectively, of the atom associated with the weight a_j. In other words, if an atom is completely in $[t,t+K)$, it contributes all of its energy to the approximation; otherwise, it contributes only a fraction based on how its support intersects $[t,t+K)$. With this we are now slightly underestimating the localized energies, as seen in Fig. 4. In both of these cases for $\{w_j\}$, however, we can assume by the energy conservation of MP [24] that as the subsequence length becomes larger our error in estimating the subsequence energy goes to zero, i.e.,

$$\lim_{K\to N_i} \|\mathbf{P}_t\mathbf{y}_i\|^2 - \sum_{j=1}^{n_t} w_j a_j^2 = \|\mathbf{P}_t\mathbf{r}_i(n_i)\|^2 \quad (26)$$

With a complete dictionary, we can make the right hand side zero. Significant departures from the energy estimate of subsequences can be due to the interactions between atoms [37].

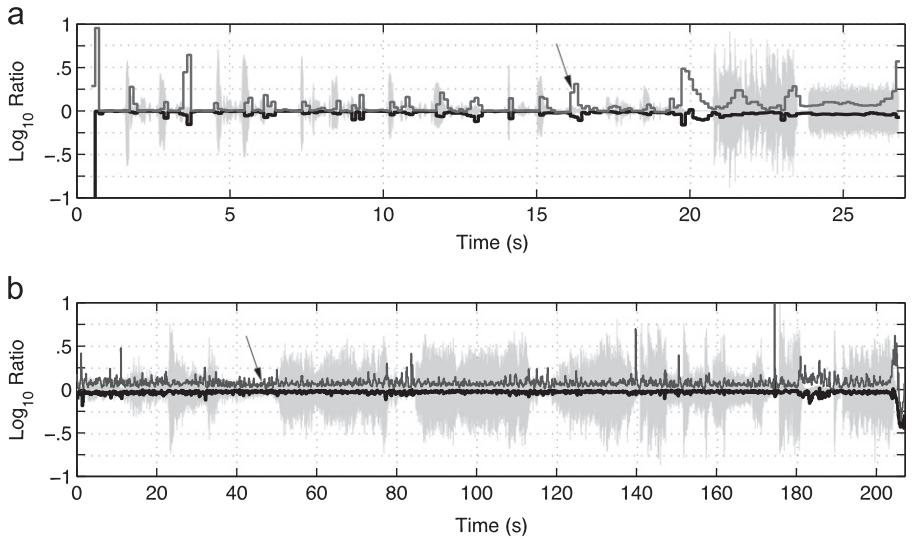

Fig. 4. Short-term energy ratios, $\log_{10}(\sum_{j=1}^{n_t} w_j a_j^2/\|\mathbf{P}_t\mathbf{y}_i\|^2)$, over 1 s windows (hopped 125 ms) for MP decompositions using 8xMDCT [31] to a global residual energy 30 dB below the initial signal energy. Arrow points to line (top, gray) using weighting $w_j=1$. The other line (bottom, black) uses (25). Data in (a) are described in Section 4.2; data in (b) are described in Section 4.3. (a) Six speech signals (0-20 s), Music except (21-23 s), Realization of GWN (24-27 s) and (b) Music Orchestra.

3.2. Recursive subsequence search limited by bounds

Now, similar to (9) and (10), we can say,

$$\langle \mathbf{x}_q, \mathbf{P}_t \mathbf{y}_i \rangle \approx \xi_t \sum_{m=1}^{n_t} \sum_{l=1}^{n_q} [\mathbf{A}_{iq}(t) \bullet \mathbf{G}_{iq}(t)]_{ml} = S_{iq}(t,M) + R(t,M) \tag{27}$$

where for $M = 2,3,\ldots,\min(n_i,n_q)$, and with $S_{iq}(t,1) = \xi_t [\mathbf{A}_{iq}(t) \bullet \mathbf{G}_{iq}(t)]_{11}$,

$$S_{iq}(t,M) \triangleq S_{iq}(t,M-1) + \xi_t \sum_{m=1}^{M} [\mathbf{A}_{iq}(t) \bullet \mathbf{G}_{iq}(t)]_{m(M-m+1)} \tag{28}$$

The problem of finding the subsequence closest to $\mathbf{x}_q$ with respect to the cosine distance can now be done iteratively over M by bounding each remainder $R(t,M)$ using (14), (15), or (16), and the method presented in Section 2.1. Furthermore, we can compare only a subset of all possible subsequences using a coarse time partition $\mathcal{T}_i$.

3.3. Practicality of the bounds for audio signals

The experiments by Jost et al. [14,15] use small images (128 square) and orthogonal wavelet decompositions, which do not translate to audio signals decomposed over redundant time–frequency dictionaries. Jost et al. [14,15] do not state the compressibility parameters they use, but for the high-dimensional audio signals with which we work in this paper it is not unusual to have $\gamma \approx 0.5$ when using MP and highly overcomplete dictionaries. We find that decomposing 4 s segments of music signals (single channel, 44.1 kHz sample rate, representation weights shown in Fig. 1) using the dictionary in Table 1 requires on average 2375 atoms to reduce the residual energy 20 dB below the initial signal energy. Thus, for the bound (16) using $n=2375$ atoms, and with the parameters $(C,\gamma) = (0.4785,0.5)$ (in the feasible set), Fig. 5 clearly shows that in order to have any discriminable bound (say ± 0.2 for unit-norm signals) we must either select a low value for p—in which case we are assuming the first atom comparison is approximately the cosine distance—or we must make over a million pairwise comparisons.

This is not practical for signals of large dimension, and dictionaries containing billions of time–frequency atoms. There is no possibility of tabulating the dictionary Gramian for quick lookup of atom pair correlations; and the cost of looking up atoms in million-atom decompositions is expensive as well. It is clear then that the tightest bound given in (16) is not practical for efficiently discriminating distances between audio signals with respect to their cosine distance (5) decomposed by MP and time–frequency dictionaries.

Table 1
Time–frequency dictionary parameters (44.1 kHz sampling rate): atom scale s, time resolution Δ_u, and frequency resolution Δ_f. Finer frequency resolution for small-scale atoms is achieved with interpolation by zero-padding.

s (samples/ms)	Δ_u (samples/ms)	Δ_f (Hz)
128/3	32/0.7	43.1
256/6	64/2	43.1
512/12	128/3	43.1
1024/23	256/6	43.1
2048/46	512/12	21.5
4096/93	1024/23	10.8
8192/186	2048/46	5.4
16,384/372	4096/93	2.7
32,768/743	8192/186	1.3

4. Experiments in comparing audio signals in a sparse domain

Though approximate nearest neighbor subsequence search of sparsely approximated audio signals with the bound (16) is impractical, we have found that approximating the cosine distance in a sparse domain has some intriguing behaviors. We now present several experiments where we compare different types of audio data in a sparse domain under a variety of conditions.

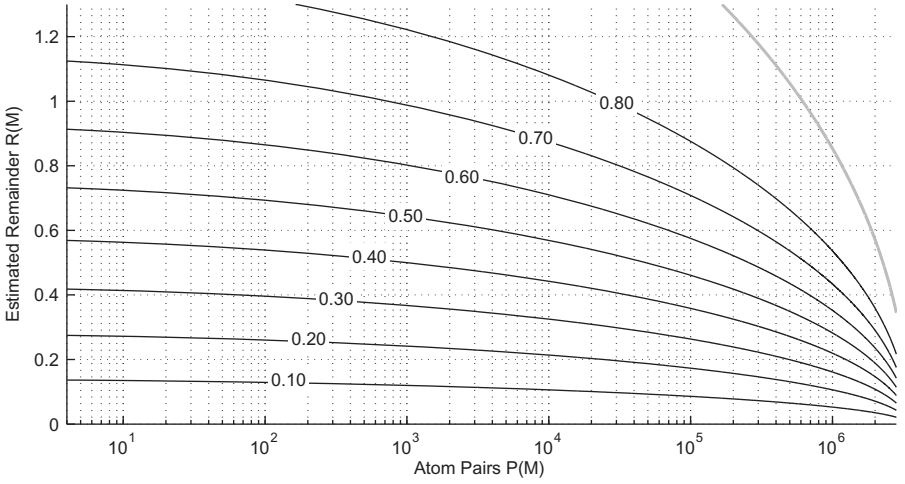

Fig. 5. Estimated remainder (assuming unit-norm signals) as a function of the number of atom pairs already considered for dataset used to produce Fig. 1. Gray: bound in (15). Black, numbered: bound in (16) for several labeled p (probability that remainder does not exceed bound) with $n=2375$ (number of elements in each sparse model), and $(C,\gamma) = (0.4785,0.5)$.

All signals are single channel, and have a sampling rate of 44.1 kHz. We decompose each by MP [17] using either the dictionary in Table 1, or the 8xMDCT dictionary [31].

4.1. Experiment 1: comparing piano notes

In this experiment, we look at how well low-order sparse approximations of sampled piano notes embody their harmonic characteristics by comparing them using the methods presented in Section 2. The data in set "A" are 68 notes (chromatically spanning A0 to G#6) on a real and somewhat in-tune piano; and in set the data "B" are 39 notes (roughly a C major scale C0 to D6) on a real and very out-of-tune piano with very poor recording conditions. We truncate all signals to have a dimension of 176,400 (4 s), and decompose each by MP [17] over a redundant dictionary of time–frequency Gabor atoms, the parameters of which are summarized in Table 1. We stop each decomposition once its residual energy drops 40 dB below the initial energy. We normalize the weights of each model by the square root energy of the respective signal. We do not align the time-domain signals such that the note onsets occur at the same time. Fig. 1 shows the ordered decays of the weights in the sparse models of data set "A".

Fig. 6(a) shows the magnitude correlations between all pairs of signals in set "A" evaluated in the time-domain.

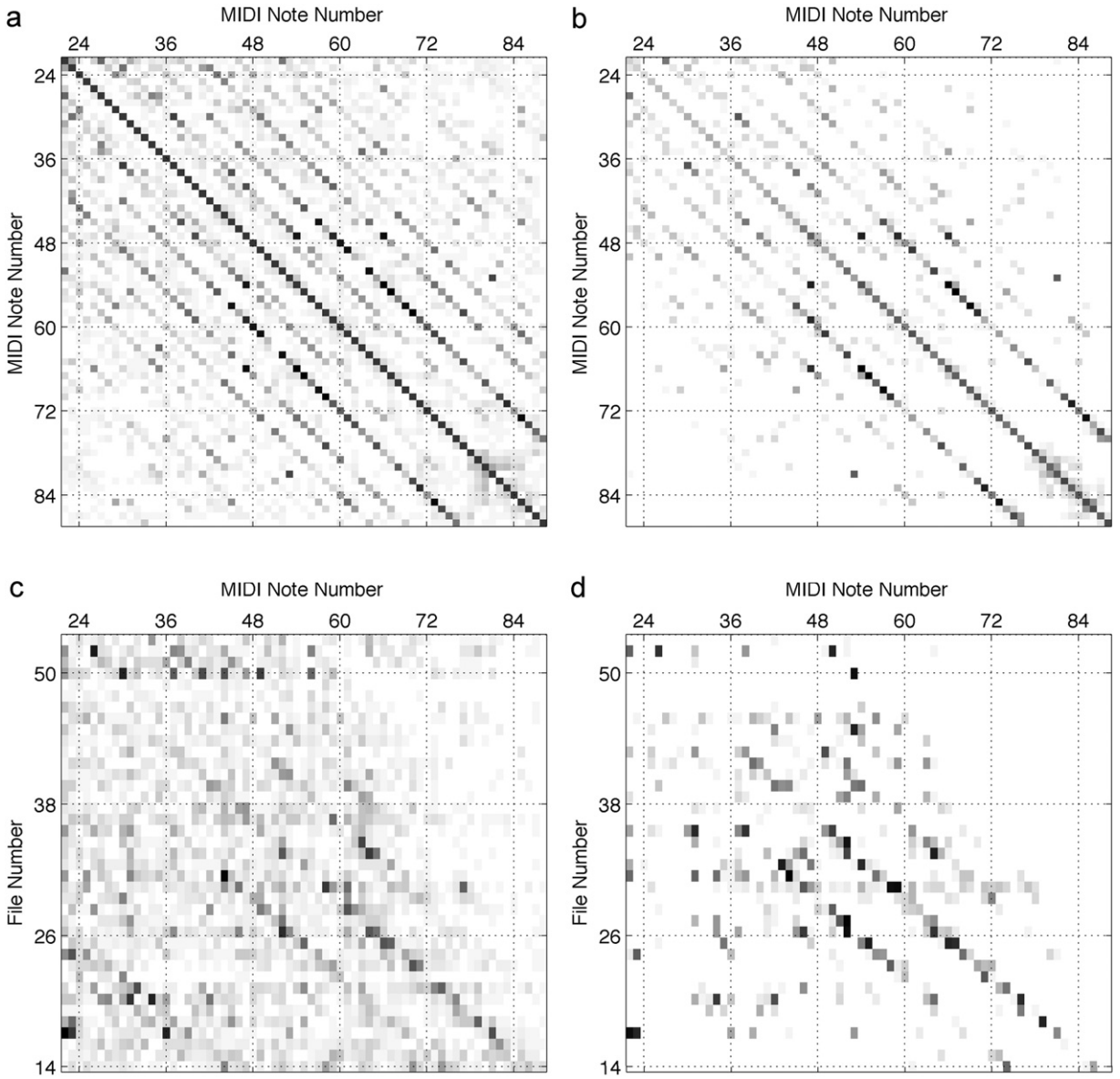

Fig. 6. $|S_{iq}(10)|$ (9) for two sets of recorded piano notes. (a) and (b): Set "A" compared with itself in time and sparse domains ($M=10$). (c) and (d): Set "B" (rows) compared with set "A" (columns) in time and sparse domains ($M=10$). Elements on main diagonals in (a) and (b) are scaled by 0.25 to increase contrast of other elements. (a) 'A' Time-domain magnitude correlations, (b)'A' Sparse-domain approximations of magnitude correlations, (c) 'B'-'A' Time-domain magnitude correlations and (d) 'B'-'A' Sparse-domain approximations of magnitude correlations.

The overtone series is clear as diagonal lines offset at 12, 19, 24, and 28, semitones from the main diagonal. Fig. 6(b) shows the approximated magnitude correlations (9) using only $M=10$ atoms from each signal approximation (thus $P(10)=55$ atom pairs). Even though the mean number of atoms in this set of models is about 7000 we still see portions of the overtone series. The diagonal in Fig. 6(b) does not have a uniform color because low notes have longer sustain times than high notes, and the sparse models thus have more time–frequency atoms with greater energies spread over the support of the signal. Fig. 6(c) show the magnitude correlations between sets "B" and "A" evaluated in the time-domain; and Fig. 6(d) shows the magnitude correlations (9) using only $M=10$ atoms from each model. In a sparse domain, we can more clearly see the relationships between the two sets because the first 10 terms of each model are most likely related to the stable harmonics of the notes, and not to the noise. We can see a diatonic scale starting around MIDI number 36, as well as the fact that the pitch scale in data set "B" lies somewhere in-between the semitones in data set "A".

Fig. 7(a) shows the approximate magnitude correlations $|S_{iq}(M)|$ (9), as well as the upper and lower bounds on the remainder using the tightest bound (16) with $p=0.2$, and $n=100$, for the signal A3 from set "A" and the rest of the set. Here we can see that the lower bound for the largest magnitude correlation exceeds the upper bound of all the rest after comparing only $M=19$ atoms from each decomposition. All but five of the signals can be excluded from the search after $M=6$. The four other signals having the largest approximate magnitude correlation are labeled, and are harmonically related to the signal through its overtone series. With a signal selected from set "B" and compared to set "A", Fig. 7(b) shows that we must compare many more atoms between the models until the bounds have any discriminability. After $P(M)=1500$ atom comparisons we can see that the largest magnitudes $|S_{iq}(M)|$ (9) are roughly harmonically related to the detuned D5 from set "B".

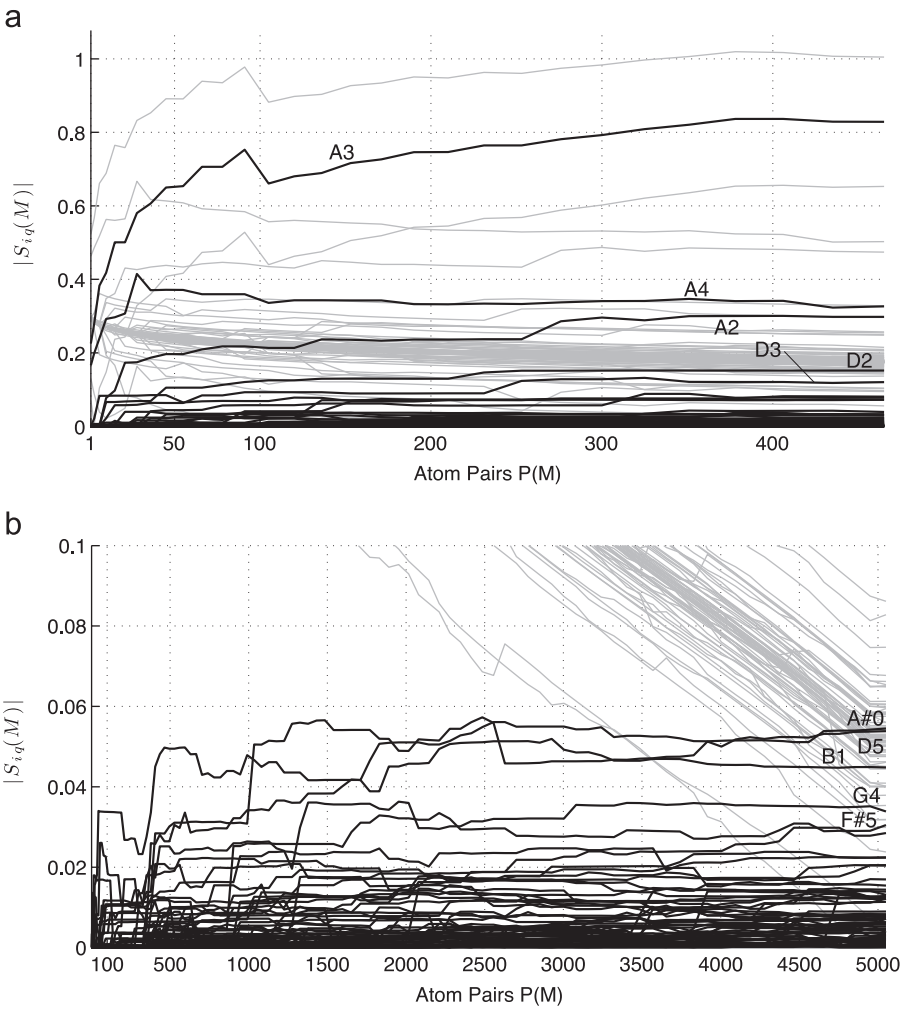

Fig. 7. Black: $|S_{iq}(M)|$ (9) as a function of the number of atom pairs considered for the set of piano notes in "A" with a signal from either (a) "A" (note A3) or (b) "B" (note D5 approximately). Gray: for each $S_{iq}(M)$, magnitudes of $L_{iq}(M)$ (12) and $U_{iq}(M)$ (13) using bound in (16) with $p=0.2$ (probability that remainder does not exceed bound), and $n=100$ (number of elements in each sparse model). Largest magnitude correlations are labeled. Note differences in axes. Signal A3 from 'A' with $(c,\gamma)=(0.78, 0.60)$ and (b) Signal D5 from 'B' with $(c, \gamma)=(1.17,0.70)$.

As a final part of this experiment, we look at the effects of comparing atoms with parameters within some subset. As done in Fig. 6(d), we compare the sparse approximations of two different sets of piano notes, but here we only consider those atoms that have scales greater than 186 ms. This in effect means that we look for signals that share the same "long-term" time–frequency behaviors. The resulting $|S_{iq}(10)|$ (9) is shown in Fig. 8. We see correlations between the notes much more clearly compared with Fig. 6(d). Removing the short-term phenomena improves "tonal"-level comparisons between the signals because non-overlapping yet energetic short atoms are replaced by atoms representative of the note harmonics.

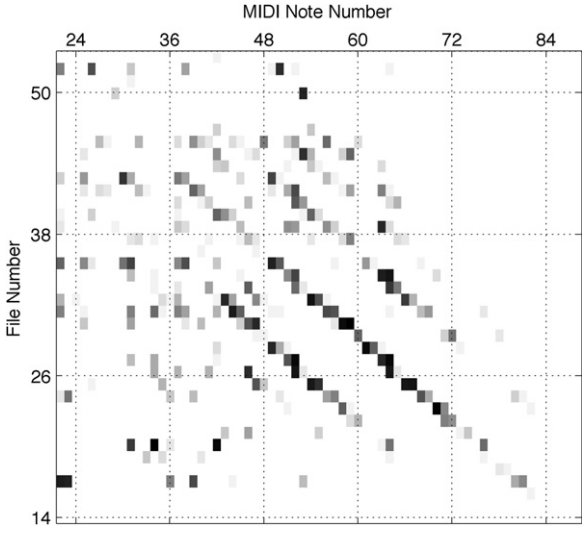

Fig. 8. $|S_{iq}(10)|$ (9) for two sets of recorded piano notes in a sparse domain using only the atoms with duration at least 186 ms. Compare with Fig. 6(d).

4.2. Experiment 2: comparing speech signals

In this experiment, we test how efficiently using (28) we can find in a speech signal the time from which we extract some $\mathbf{x}_q$. We also test how distortion in the query affects these results. We make a signal by combining six segments of speech, a short music segment, and white noise, shown in Fig. 4(a). The six speech segments are the same phrase spoken by three females and three males: "Cottage cheese with chives is delicious." We extract from one of these speech signals the word "cheese," to create $\mathbf{x}_q$ with duration of 603 ms, shown at top in Fig. 9. We decompose this signal using MP and the 8xMDCT dictionary [31].

We distort the query in two ways: with additive WGN (AWGN), and with an interfering sound having a high correlation with the dictionary. In the first case, shown in the middle in Fig. 9, the signal $\mathbf{x}_q' = (\alpha\mathbf{x}_q + \mathbf{n})/\|\alpha\mathbf{x}_q + \mathbf{n}\|$ is the original $\mathbf{x}_q$ distorted by a unit-norm AWGN signal $\mathbf{n}$. We set $\alpha = 0.3162$ such that $10\log_{10}(\|\alpha\mathbf{x}_q\|^2/\|\mathbf{n}\|^2) = 20\log_{10}(|\alpha|) = -10$ dB. For this signal, we find the following statistics from 10,000 realizations of the AWGN signal: $\mathrm{E}[|\langle\mathbf{x}_q,\mathbf{n}\rangle|] \approx 1\times10^{-5}$, $\mathrm{Var}[|\langle\mathbf{x}_q,\mathbf{n}\rangle|] \approx 4\times10^{-6}$. We also find the following statistics for the 8xMDCT

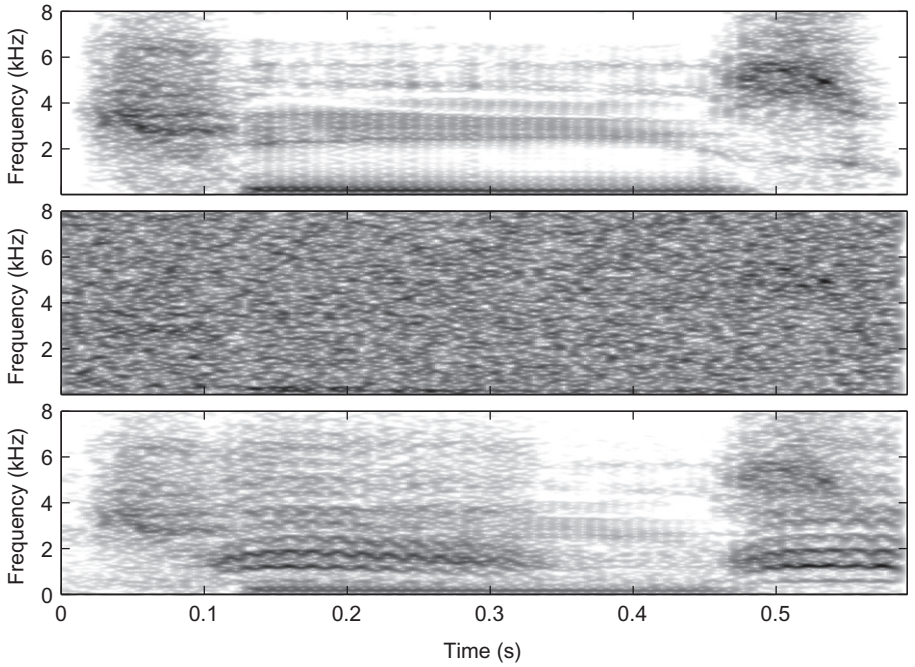

Fig. 9. Log spectrograms of the query signals with which we search. Top: query of male saying "cheese." Middle: query distorted with additive white Gaussian noise (AWGN) with SNR $= -10$ dB. Bottom: query distorted with interfering crow sound with SNR $= -5$ dB.

dictionary: $\mathrm{E}[\max_{\mathbf{d}\in\mathcal{D}}|\langle\mathbf{n},\mathbf{d}\rangle|]\approx 5\times10^{-4}$, $\mathrm{Var}[\max_{\mathbf{d}\in\mathcal{D}}|\langle\mathbf{n},\mathbf{d}\rangle|]\approx 2\times10^{-5}$. Thus, the noise signal is not well correlated either with the original signal or the 8xMDCT dictionary. In the second case, shown at the bottom of Fig. 9, we distort the signal by adding the sound of a crow $\mathbf{c}$ so that $\mathbf{x}_q'=(\alpha\mathbf{x}_q+\mathbf{c})/\|\alpha\mathbf{x}_q+\mathbf{c}\|_2$ with $\|\mathbf{c}\|=1$. Here, we set $\alpha=0.5623$ given by $20\log_{10}(|\alpha|)=-5$ dB. For this interfering signal, we find that $|\langle\mathbf{x}_q,\mathbf{c}\rangle|\approx 2\times10^{-3}$, but $\max_{\mathbf{d}\in\mathcal{D}}|\langle\mathbf{c},\mathbf{d}\rangle|\approx 0.21$, which is higher than $\max_{\mathbf{d}\in\mathcal{D}}|\langle\mathbf{x}_q,\mathbf{d}\rangle|\approx 0.17$. In this case, unlike for the AWGN interference, it is likely that the sparse approximation of the signal with the crow interference will have atoms in its low-order model due to the crow and not the speech. We do not expect the AWGN interference to be a part of the signal model created by MP until much later iterations.

Fig. 10 shows $|S_{iq}(t,M)|$ (28) aligned with the original signal for four values of M using the sparse approximations of the clean and distorted signals. We plot at the rear of these figures the localized magnitude time-domain correlation of the windowed and normalized signal with the query $\mathbf{x}_q$. In Fig. 10(a), using the clean $\mathbf{x}_q$, we clearly see its position even when using a single atom pair for each 100 ms partition of the time-domain. We see the same behavior in Fig. 10(b), (c) for the two distorted signals, but in the case where the crow sound interferes we find the query for $M\geq 2$, or with at least three atom pair comparisons. The first atom of the decomposed query with the crow is modeling the crow and not the content of interest, and so we must increase the order of the model to find the location of $\mathbf{x}_q$. As we increase the number of pairs considered we also find other segments that point in the same direction as $\mathbf{x}_q$. Table 2 gives the times and content of the ten largest values in $|S_{iq}(t,10)|$. For the clean and AWGN distorted $\mathbf{x}_q$, "cheese" appears five of the six times it exists in the original signal. Curiously, these same five instances are the five largest magnitude correlations when $\mathbf{x}_q$ has the crow interference.

We perform the same test as above but using a much longer speech signal (about 21 minutes in length) excerpted from a book-on-CD, "The Old Man and the Sea" by Ernest Hemingway, read aloud by a single person. From this signal we create several queries $\mathbf{x}_q$, from words

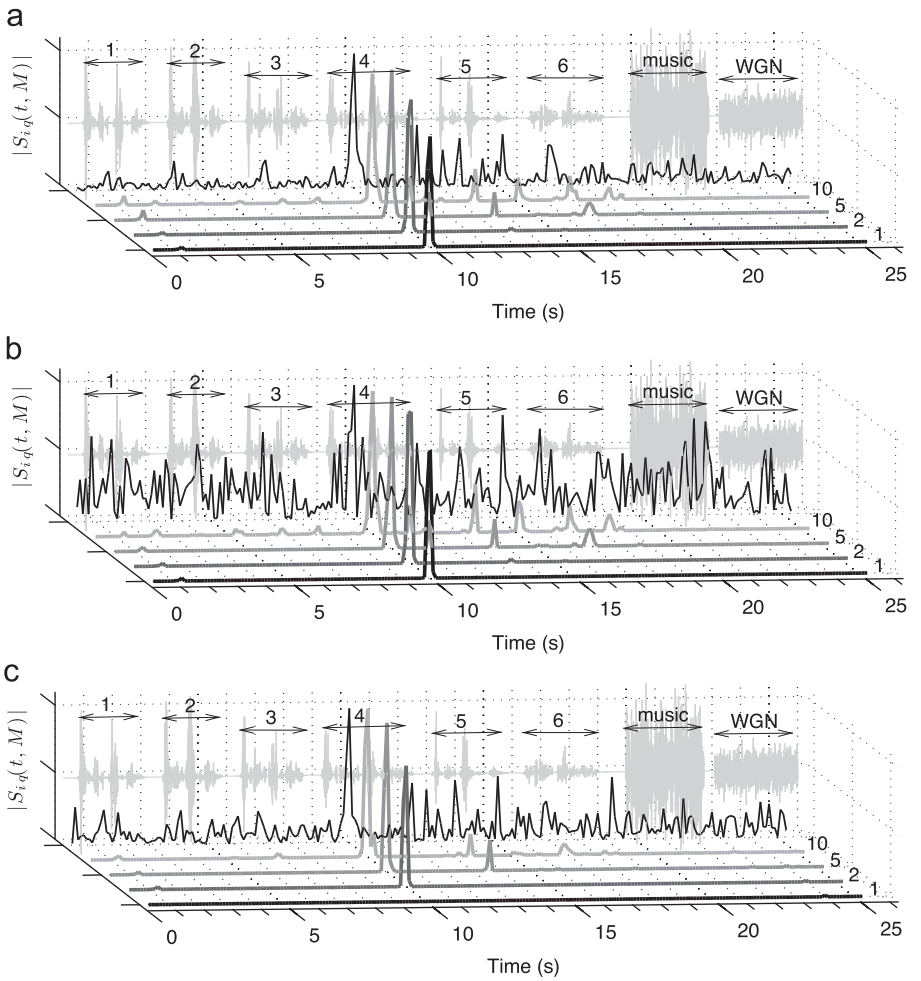

Fig. 10. $|S_{iq}(M,t)|$ (28) as a function of time and the number of atoms M (labeled at right) considered from each representation for each localized sparse approximation. Localized magnitude correlation of each signal with query is shown by the thin black line in front of the gray time-domain signal at rear. (a) clean signal, (b) Signal with AWGN at 10 dB energy and (c) Signal with crow signal at 5 dB energy.

Table 2
Times, values and signal content for first 10 peaks in $|S_{iq}(t,10)|$ ($P(10)=55$) in Figs. 10(a)–(c). Highest-rated distances in each (bold) points to the origin of signal.

#	Clean signal			Signal + WGN			Signal + Crow		
	t (s)	$\|S_{iq}\|$	Content	t (s)	$\|S_{iq}\|$	Content	t (s)	$\|S_{iq}\|$	Content
1	**10.0**	**0.798**	"**cheese**"	**10.0**	**0.236**	"**cheese**"	**10.0**	**0.409**	"**cheese**"
2	13.6	0.199	"cheese"	13.6	0.080	"cheese"	13.6	0.060	"cheese"
3	11.3	0.153	"-ives is-"	15.1	0.051	"delicious"	16.9	0.030	"cheese"
4	16.9	0.149	"cheese"	11.3	0.045	"-ives is-"	6.9	0.012	"cheese"
5	15.1	0.141	"delicious"	16.9	0.042	"cheese"	1.3	0.011	"cheese"
6	18.3	0.076	"delicious"	18.3	0.028	"delicious"	18.3	0.010	"delicious"
7	1.3	0.057	"cheese"	8.1	0.014	"delicious"	13.2	0.010	"cottage"
8	8.1	0.035	"delicious"	12.0	0.012	"-licious"	15.1	0.009	"delicious"
9	2.4	0.026	"delicious"	5.2	0.011	"delicious"	16.0	0.004	"cott-"
10	6.9	0.024	"cheese"	6.8	0.010	"cheese"	22.8	0.003	WGN

to sentences to an entire paragraph of duration 35 s. We decompose each signal over the dictionary in Table 1 until the global residual energy is 20 dB below the initial energy. The approximation of the entire 21 m signal has 1,004,001 atoms selected from a dictionary containing 2,194,730,297 atoms.

One $\mathbf{x}_q$ we extract from the signal is the spoken phrase, "the old man said" (861 ms in length). This phrase appears 26 times in the long excerpt. We evaluate $|S_{iq}(t,M)|$ (28) every 116 ms, and find the time at which $\mathbf{x}_q$ originally appears using only $M=1$ atom pair comparisons for each time partition. The next highest ranked positions have values of 75% and 67% that of the largest $|S_{iq}(t,1)|$. When $M=50$, the values of the second and third largest values $|S_{iq}(t,50)|$ drop to 62% and 61% that of the largest value. In the top 30 ranked subsequences for $M=5$ we find only one of the other 25 appearances of "the old man said" (rank 26); but we also find "the old man agreed" (rank 11), and "the old man carried" (rank 16). All other results have minimal content similarity to the signal, but have time–frequency overlap in parts of the atoms of each model.

We perform the same test with a sentence extracted from the signal, "They were as old as erosions in a fishless desert" (2.87 s), which only appears once. No matter the $M=[1, 50]$ we use, the origin of the excerpt remains at a rank of 6 with a value $|S_{iq}(t,50)|$ at 67.5% that of the highest rank subsequence. We find that if we shift the time partition forward by 11.6 ms its ranking jumps to first, with the second ranked subsequence at 73%. We observe a similar effect for a query consisting of an entire paragraph (35 s). We find its origin by comparing $M=2$ or more atoms from each model using a time partition of 116 ms. This result, however, disappears when we evaluate $|S_{iq}(t,M)|$ using a coarser time partition of 250 ms.

4.3. Experiment 3: comparing music signals

While the previous experiment deals with single-channel speech signals, in this experiment we make comparisons between polyphonic musical signals excerpted from a commercial recording of the fourth movement of *Symphonie Fantastique* by H. Berlioz. For the query, we use a 10.31 s segment of the third appearance of the "A" theme of the movement (bars 33–39, located around 13–22 s in Fig. 4(b)). Fig. 11 shows the sonogram and time–frequency tiles of the model of $\mathbf{x}_q$ using the 50 atoms with the largest magnitude weights selected from the 8xMDCT dictionary [31]. We add no interfering signals as we do in the previous experiment.

Fig. 12(a) shows $|S_{iq}(t,M)|$ (28) over the first minute of the original signal, for three values of M, including $M=50$, the time–frequency representation of which is shown at bottom of Fig. 11. For $|S_{iq}(t,50)|$ we can see a strong spike located around 13 s corresponding with the query, but we also see spikes at about 2 s and around 43 s. The former set of spikes correspond with the second appearance of the "A" theme, when only low bowed strings are playing the theme in G-minor. This is quite similar to the instrumentation of the query: low bowed strings and a legato bassoon in counterpoint in E♭-major. The latter set of spikes is around the end of the fifth appearance of the theme, which is played in G-minor on low pizzicato strings with a staccato bassoon. For $M=10$, we see a conspicuous spike at the time of the fifth appearance around 34 s, as well as of the fourth appearance around 24 s, where the theme is played in E♭-major like the query. Finally, we test how the sparse approximation of this query compares with subsequences from a different recording of this movement, which is also in a different tempo. Fig. 12(b) shows $|S_{iq}(t,M)|$ (28) for three different values of M. We see high similarity with the first and second appearances of the main theme, but not the third, which is what the query contains.

4.4. Discussion

There is no reason to believe that a robust and accurate speech or melody recognition system can be created by comparing only the first few elements of greedy decompositions in time–frequency dictionaries. What appears to be occurring for the short signals, both the "cheese" and "the old man said," is that the first few elements of their sparse and atomic decomposition create a prosodic

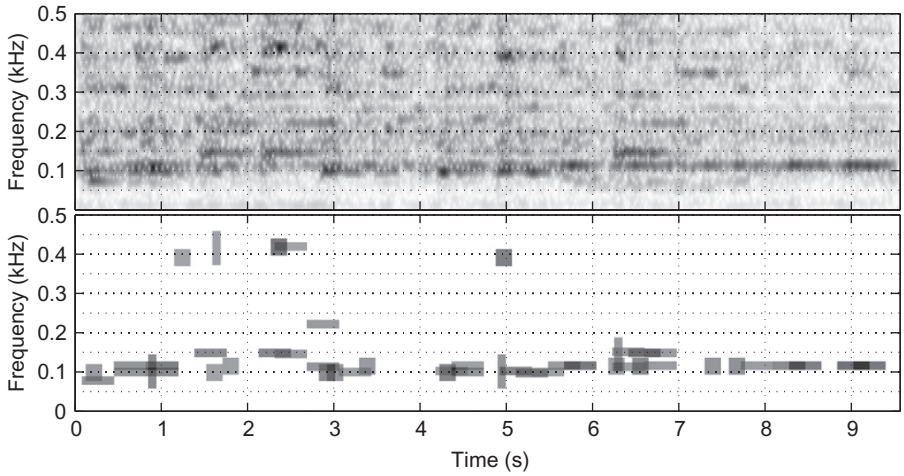

Fig. 11. Polyphonic orchestral query: sonogram (top) and time–frequency tiles (bottom) of 50-order sparse approximation.

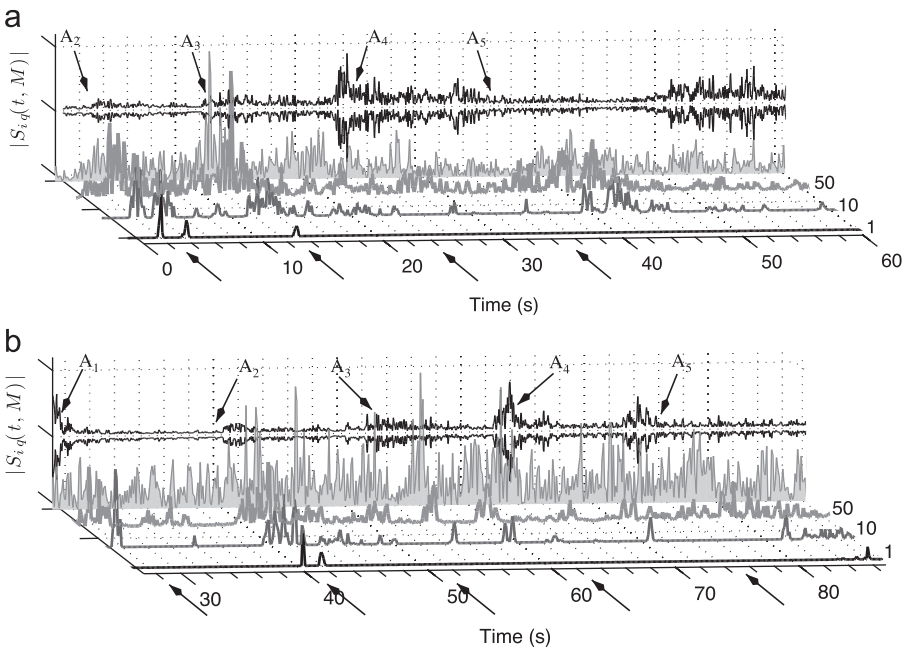

Fig. 12. $|S_{iq}(t,M)|$ (28) for three values of M for the query and two different signals. Arrows mark the appearances of the "A" theme, and their appearance number. Magnitude correlation of query with localized and normalized signal is shown by the solid gray area in front of the black time-domain signal at rear. (a) Query compared with original signal and (b) Query compared different interpretation.

representation that is comparable to others at the atomic level. For the longer signals, such as sentences, paragraphs, and an orchestral theme, a few atoms cannot adequately embody the prosody, but we still see that by only making a few comparisons we are able to locate the excerpted signal—as long as the time partition is fine enough. This is due to the atoms of the models acting in some sense as a time–frequency fingerprint, an example of which we see in Fig. 11. Through the cosine distance, the relative time and frequency locations of the atoms in the query and subsequence are being compared, weighted by their energies. Subsequences that share atoms in similar configurations will be gauged closer to $\mathbf{x}_q$ than those that do not.

By using the cosine distance it is not unexpected that (28) will be extremely sensitive to a partitioning of the time-domain. This comes directly from the definition of the time-localized Gramian (23), as well as the use of a dictionary that is not translation invariant. There is no need to partition the time axis when using a parameterized dictionary if we assume that some of the atoms in the model of $\mathbf{x}_q$ will have parameters that are nearly the same as some of those in the

relevant localized sparse representations. In such a scenario, we can search a sparse representation for the times at which atoms exist that are similar in scale and modulation frequency to those modeling $\mathbf{x}_q$. Then we can limit our search to those particular times without considering any uniform and arbitrary partition of the time-domain. With non-linear greedy decomposition methods such as MP and time-variant dictionaries, however, such an assumption cannot be guaranteed; but its limits are not yet well-known.

5. Conclusion

In this paper, we have extended and investigated the applicability of a method of recursive nearest neighbor search [14,15] for comparing audio signals using pairwise comparisons of model elements in a sparse domain. The multiscale descriptions offered by sparse approximation over time–frequency dictionaries are especially attractive for such tasks because they provide flexibility in making comparisons between data, not to mention a capacity to deal with noise. After extending this method to the task of comparing subsequences of audio signals, we find that the strongest bound known for the remainder is too weak to quickly and efficiently reduce the search space. Our experiments show, however, that by comparing elements of sparse models we can judge with relatively few comparisons whether signals share the same time–frequency structures, and to what degrees, although this can be quite sensitive to the time-domain partitioning. We also see that we can approach such comparisons hierarchically, starting from the most energetic content to the least, or starting from the longest scale phenomenon to the shortest.

We are continuing this research in multiple directions. First, since we know that the inner product matrix $\mathbf{G}_{iq}(t)$ (23) will be very sparse for all t in time–frequency dictionaries, this motivates designing a tighter bound based on a Laplacian distribution of elements in $\mathbf{G}_{iq}(t)$ with a large probability mass exactly at zero. This bound would be much more realistic than that provided by assuming the elements of the Gramian are distributed uniform (16). Another part of the problem is of course that the sums in (9) and (28) are not such that at step M the $P(M)$ largest magnitude values of $\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}$ are actually being summed. By our assumption in (4), we know that the decay of the magnitudes of the elements in $\mathbf{A}_{iq}$ will be quickest in diagonal directions, but dependent upon the element position in the matrix. These diagonal directions are simply given by

$$\begin{bmatrix} \partial/\partial\gamma_i \\ \partial/\partial\gamma_q \end{bmatrix} m^{-\gamma_i} l^{-\gamma_q} = -m^{-\gamma_i} l^{-\gamma_q} \begin{bmatrix} \gamma_i/m \\ \gamma_q/l \end{bmatrix} \tag{29}$$

where we now recognize that the weights of two different representations can decay at different rates. With this, we can make an ordered set of index pairs by

$$\Lambda = \{(m,l)_\lambda : |[\mathbf{A}_{iq}]_\lambda| \geq |[\mathbf{A}_{iq}]_{\lambda+1}|\}_{\lambda=1,2,\ldots,n_i n_q} \tag{30}$$

and define a recursive sum for $1 < m \leq n_i n_q$

$$S_{iq}(m) \triangleq S_{iq}(m-1) + [\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{\Lambda_m} \tag{31}$$

setting $S_{iq}(1) = [\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}]_{11}$. We do not yet know the extent to which this approach can ameliorate the problems with the non-discriminating bound (16), as we have yet to design an efficient way to generate a satisfactory Λ, and estimate the bounds of the corresponding remainder—whether it is like that in (16), or another that uses the fact that $\mathbf{G}_{iq}(t)$ will be very sparse, even when $\mathbf{x}_q = \mathbf{y}_i$. We think that using a stronger bound and this indexing order will significantly reduce the number of pairwise comparisons that must be made before determining a subsequence is not close enough with respect to the cosine distance. Furthermore, we can make the elements of $\mathbf{A}_{iq}$ decay faster, and thus increase γ, by using other sparse approximation approaches, such as OMP [28,23] or CMP [36]. And we cannot forget the implications of choosing a particular dictionary. In this work, we have used two different parametric dictionaries, one of which is designed for audio signal coding [31]. Another interesting research direction is to use dictionaries better suited for content description than coding, such as content-adapted dictionaries [20,2,19].

Finally, and specifically with regards to the specific problem of similarity search in audio signals, the cosine distance between time-domain samples makes little sense because it is too sensitive to signal waveforms whereas human perception is not. Instead, many other possibilities exist for comparing sparse approximation, such as comparing low-level histograms of atom parameters [7,34]; comparing mid-level structures such as harmonics [9,38,8]; and comparing high-level patterns of short atoms representing rhythm [32]. There also exists the matching pursuit dissimilarity measure [25], where the atoms of one sparse model are used to decompose another signal, and vice versa to see how well they model each other. We are exploring these various possibilities with regards to gauging more generally similarity in audio signals at multiple levels of specificity within a sparse domain.

Acknowledgments

B.L. Sturm performed part of this research as a Chateaubriand Post-doctoral Fellow (N. 634146B) at the Institut Jean Le Rond d'Alembert, Équipe Lutheries, Acoustique, Musique, Université Pierre et Marie Curie, Paris 6, France; as well as at the Department of Architecture, Design and Media Technology, at Aalborg University Copenhagen, Denmark. L. Daudet acknowledges partial support from the French Agence Nationale de la Recherche under contract ANR-06-JCJC-0027-01 DESAM. The authors thank the anonymous reviewers for their very detailed and helpful comments.

Appendix A. Proof of remainder bounds

To show (14), we can bound $R(M)$ loosely by assuming the worst case scenario of $[\mathbf{G}_{iq}]_{ml} = 1$ for all its elements. Knowing that $R(M)$ is the sum of the elements of the matrix $\mathbf{A}_{iq}\bullet\mathbf{G}_{iq}$ except for the first $P(M)$ values, and assuming (4), we can say

$$\begin{aligned} C^{-2} R(M) &\leq \sum_{m=M+1}^{n} \sum_{l=1}^{m} [l(m-l+1)]^{-\gamma} + \sum_{m=1}^{n-1} \sum_{l=m+1}^{n} [l(n-m+1)]^{-\gamma} \\ &= \|\mathbf{c}_M^{\gamma}\|_1 + \|\mathbf{d}^{\gamma}\|_1 \end{aligned} \tag{A.1}$$

where $\mathbf{c}_M^\gamma$ and $\mathbf{d}^\gamma$ are defined in (17) and (18). This worst case scenario is not possible using MP because of its update rule (1).

We can find the tighter bound in (15) by assuming the distribution of signs of the elements of $\mathbf{G}_{iq}$ is Bernoulli equiprobable, i.e., $P\{[\mathbf{G}_{iq}]_{ml}=1\}=P\{[\mathbf{G}_{iq}]_{ml}=-1\}=0.5$. Thus, defining a random variable $b_i:\mathbb{R}\mapsto\{-1,1\}$, and its probability mass function $f_B(b_i)=0.5\delta(b_i+1)+0.5\delta(b_i-1)$ using the Dirac function, $\delta(x)$, we create a random vector $\mathbf{b}$ with $n^2-P(M)$ elements independently drawn from this distribution. Placing this into the double sums of (A.1) provides the bound

$$C^{-2}R(M)\le\left|\mathbf{b}^T\begin{bmatrix}\mathbf{c}_M^\gamma\\ \mathbf{d}^\gamma\end{bmatrix}\right|\le\|\mathbf{c}_M^\gamma\|_1+\|\mathbf{d}^\gamma\|_1 \tag{A.2}$$

This weighted Rademacher sequence has the property that [14]

$$P\{|\mathbf{b}^T\mathbf{s}|>R\}\le 2\exp(-R^2/2\|\mathbf{s}\|_2^2),\quad R>0 \tag{A.3}$$

which becomes $P\{|\mathbf{b}^T\mathbf{s}|\le R\}\ge\max\{0,1-2\exp(-R^2/2\|\mathbf{s}\|_2^2)\}$ by the axioms of probability. With this we can find an R such that $P\{|\mathbf{b}^T\mathbf{s}|\le R\}$ will be greater than or equal to some probability $0\le p\le 1$, i.e.,

$$R(p)=(\|\mathbf{c}_M^\gamma\|_2^2+\|\mathbf{d}^\gamma\|_2^2)^{1/2}\left[2\ln\frac{2}{1-p}\right]^{1/2} \tag{A.4}$$

This value can be minimized by choosing $p=0$, for which we arrive at the residual upper bound in (15). Note that even though we have set $p=0$, we still have an unrealistically loose bound by the impossibility of MP of choosing two sets of atoms for which all entries of their Gramian $\mathbf{G}_{iq}$ are in $\{-1,1\}$.

Finally, to show (16), we can model the elements of the Gramian as random variables, $u_i:\mathbb{R}\mapsto[-1,1]$, independently and identically distributed uniformly

$$f_U(u_i)=\begin{cases}0.5, & -1\le u_i\le 1\\ 0 & \text{else}\end{cases} \tag{A.5}$$

Substituting this into (14) gives a weighted sum of random variables satisfying

$$C^{-2}R(M)\le\left|\mathbf{u}^T\begin{bmatrix}\mathbf{c}_M^\gamma\\ \mathbf{d}^\gamma\end{bmatrix}\right|\le\|\mathbf{c}_M^\gamma\|_1+\|\mathbf{d}^\gamma\|_1 \tag{A.6}$$

where $\mathbf{u}$ is the random vector. For large M, this sum has the asymptotic property [14,15]:

$$P\{|\mathbf{u}^T\mathbf{s}|<R\}=\operatorname{Erf}\sqrt{\frac{3R^2}{2\|\mathbf{s}\|_2^2}} \tag{A.7}$$

Setting this equal to $0\le p\le 1$ and solving for R produces the upper bound (16). We can reach the upper bound (15) if we set $p=0.9586$, but note that (16) can be made zero. This bound can still be extremely loose because the Gramian of two models in time–frequency dictionaries will be highly sparse.

Computing the ℓ_2-norm in these expressions, however, leads to evaluating the double sums

$$\|\mathbf{c}_M^\gamma\|^2=\sum_{m=M+1}^{n}\sum_{l=1}^{m}\frac{1}{[l(m-l+1)]^{2\gamma}} \tag{A.8}$$

$$\|\mathbf{d}^\gamma\|^2=\sum_{m=1}^{n-1}\sum_{l=m+1}^{n}\frac{1}{[l(n-m+1)]^{2\gamma}} \tag{A.9}$$

which can be prohibitive for large n. The dimensionality of $\mathbf{c}_M^\gamma$ is $n(n+1)/2-P(M)$, and of $\mathbf{d}^\gamma$ is $n(n-1)/2$. We approximate these values in the following way for $\gamma=0.5$, using the partial sum of the harmonic series

$$\sum_{m=1}^{n}\frac{1}{m}=\ln n+\gamma_E+\frac{1}{2n}-\frac{1}{12n^2}+\frac{1}{120n^4}+\mathcal{O}(n^{-6}) \tag{A.10}$$

where $\gamma_E\approx 0.5772$ is the Euler–Mascheroni constant. To find $\|\mathbf{d}^{0.5}\|^2$

$$\begin{aligned}\|\mathbf{d}^{0.5}\|^2&=\sum_{m=1}^{n-1}\sum_{l=m+1}^{n}\frac{1}{l(n-m+1)}\\ &=\sum_{m=1}^{n-1}\frac{1}{n-m+1}\left[\sum_{l=1}^{n}\frac{1}{l}-\sum_{l=1}^{m}\frac{1}{l}\right]\\ &\approx\sum_{m=1}^{n-1}\frac{1}{n-m+1}\left[\ln n/m-\frac{n-m}{2nm}+\frac{n^2-m^2}{12n^2m^2}\right].\end{aligned} \tag{A.11}$$

To find $\|\mathbf{c}_M^{0.5}\|^2$ we first use partial fractions and then the partial sum of the harmonic series:

$$\begin{aligned}\|\mathbf{c}_M^{0.5}\|^2&=\sum_{m=M+1}^{n}\sum_{l=1}^{m}\frac{1}{l(m-l+1)}\\ &=\sum_{m=M+1}^{n}\frac{1}{m+1}\sum_{l=1}^{m}\frac{1}{l}+\frac{1}{m-l+1}\\ &\approx\sum_{m=M+1}^{n}\frac{1}{m+1}\left[\ln m+\gamma_E+\frac{1}{2m}-\frac{1}{12m^2}+\sum_{l=1}^{m}\frac{1}{l}\right]\\ &\approx 2\sum_{m=M+1}^{n}\frac{1}{m+1}\left(\ln m+\gamma_E+\frac{1}{2m}-\frac{1}{12m^2}\right).\end{aligned} \tag{A.12}$$

With these expressions we can avoid double sums in calculating the bounds.

Appendix B. Estimating the compressibility parameters

We estimate the compressibility parameters (C,γ) for all signals from the entire set of representation weights. Since by (4) the parameters (C,γ) bound from above the decay of all the ordered weights, only the largest magnitude weights matter for their estimation. Thus, we define a vector, $\mathbf{a}$, of the largest n magnitude weights from each row in the set $\{\{\mathbf{a}_i(n_i)\}_{i\in\mathcal{I}},\mathbf{a}_q(n_q)\}$, which is equivalent to taking the largest weights at each approximation order. Good compressibility parameters can be given by

$$\min_{C,\gamma}\|C\mathbf{z}^\gamma-\mathbf{a}\|^2+\lambda C\quad\text{subject to } C\mathbf{z}^\gamma\succcurlyeq\mathbf{a} \tag{B.1}$$

where we define $\mathbf{z}^\gamma\triangleq[1,1/2^\gamma,\ldots,1/n^\gamma]^T$, and add a multiple of C in order to keep it from getting too large since the bounds (14)–(16) are all proportional to it. The constraint is added to ensure all elements of the difference $C\mathbf{z}^\gamma-\mathbf{a}_i$ are positive such that (4) is true.

To remove the γ component from the exponent, and since all of the elements of $\mathbf{z}$ and $\mathbf{a}$ are positive and nonzero, we can instead solve the problem

$$\min_{C,\gamma}\|\ln C\mathbf{1}+\gamma\ln\mathbf{z}-\ln\mathbf{a}\|^2+\lambda\ln C$$

$$= \min_{C,\gamma} [(\ln C)^2 n + \gamma^2 \|\ln \mathbf{z}\|^2 + \|\ln \mathbf{a}\|^2 + \lambda \ln C + 2\gamma (\ln \mathbf{z})^T (\ln C \mathbf{1} - \ln \mathbf{a}) - 2 \ln C (\ln \mathbf{a})^T \mathbf{1}] \quad \text{(B.2)}$$

subject to the constraint $C\mathbf{z}^\gamma \succcurlyeq \mathbf{a}$. Taking the partial derivative of this with respect to γ and C, we find

$$\gamma_o = \frac{(\ln \mathbf{z})^T (\ln \mathbf{a} - \ln C \mathbf{1})}{\|\ln \mathbf{z}\|^2} \quad \text{(B.3)}$$

$$C_o = \exp\left[\lambda + \frac{1}{n} \sum_{i=1}^{n} [\ln \mathbf{a} - \gamma \ln \mathbf{z}]_i\right] \quad \text{(B.4)}$$

Starting with some initial value of C then, we use the following iterative method:

1. solve for γ given a C in (B.3);
2. find the new C in (B.4) using this γ;
3. set $C' = \exp[\max(\ln \mathbf{a} - \gamma_o \ln \mathbf{z})]$ and evaluate the error $\|C'\mathbf{z}^\gamma - \mathbf{a}\|^2$;
4. repeat until the error begins to increase.

The factor λ in effect controls the step size for convergence. A typical value we use is $\lambda = \pm 0.03$ based on experiments (the sign of which depends on if the objective function decreases with decreasing C).

References

[1] R. Agrawal, C. Faloutsos, A. Swami, Efficient similarity search in sequence databases, in: Proceedings of the International Conference of Foundations of Data Organization and Algorithms, Chicago, IL, October 1993, pp. 69–84.

[2] M. Aharon, M. Elad, A. Bruckstein, K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation, IEEE Transactions of Signal Processing 54 (11) (2006) 4311–4322.

[3] M. Casey, C. Rhodes, M. Slaney, Analysis of minimum distances in high-dimensional musical spaces, IEEE Transactions on Audio, Speech and Language Processing 16 (5) (2008) 1015–1028.

[4] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, M. Slaney, Content-based music information retrieval: current directions and future challenges, Proceedings of the IEEE 96 (4) (2008) 668–696.

[5] K. Chang, J.-S.R. Jang, C.S. Iliopoulos, Music genre classification via compressive sampling, in: Proceedings of the International Society for Music Information Retrieval, Amsterdam, The Netherlands, August 2010, pp. 387–392.

[6] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, SIAM Journal Scientific Computing 20 (1) (1998) 33–61.

[7] S. Chu, S. Narayanan, C.-C.J. Kuo, Environmental sound recognition with time–frequency audio features, IEEE Transactions on Audio, Speech and Language Processing 17 (6) (2009) 1142–1158.

[8] C. Cotton, D.P.W. Ellis, Finding similar acoustic events using matching pursuit and locality-sensitive hashing, in: Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, Mohonk, NY, October 2009, pp. 125–128.

[9] L. Daudet, Sparse and structured decompositions of signals with the molecular matching pursuit, IEEE Transactions on Audio, Speech and Language Processing 14 (5) (2006) 1808–1816.

[10] D.P.W. Ellis, G.E. Poliner, Identifying 'cover songs' with chroma features and dynamic programming beat tracking, in: Proceedings of the International Conference on Acoustics, Speech and Signal Processing Honolulu, Hawaii, April 2007, pp. 1429–1432.

[11] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, Fast subsequence matching in time-series databases, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, Minneapolis, MN, 1994, pp. 419–429.

[12] J. Gemmeke, L. ten Bosch, L. Boves, B. Cranen, Using sparse representations for exemplar based continuous digit recognition, in: Proceedings of the European Signal Processing Conference. Glasgow, Scotland, August 2009, pp. 1755–1759.

[13] J. Haitsma, T. Kalker, A highly robust audio fingerprinting system with an efficient search strategy, Journal of New Music Research 32 (2) (2003) 211–221.

[14] P. Jost, Algorithmic aspects of sparse approximations, Ph.D. Thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, June 2007.

[15] P. Jost, P. Vandergheynst, On finding approximate nearest neighbours in a set of compressible signals, in: Proceedings of the European Signal Processing Conference, Lausanne, Switzerland, August 2008, pp. 1–5.

[16] A. Kimura, K. Kashino, T. Kurozumi, H. Murase, A quick search method for audio signals based on piecewise linear representation of feature trajectories, IEEE Transactions on Audio, Speech and Language Processing 16 (2) (2008) 396–407.

[17] S. Krstulovic, R. Gribonval, MPTK: Matching pursuit made tractable, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, Toulouse, France, April 2006, pp. 496–499.

[18] F. Kurth, M. Müller, Efficient index-based audio matching, IEEE Transactions on Audio, Speech and Language Processing 16 (2) (2008) 382–395.

[19] P. Leveau, E. Vincent, G. Richard, L. Daudet, Instrument-specific harmonic atoms for mid-level music representation, IEEE Transactions on Audio, Speech and Language Processing 16 (1) (2008) 116–128.

[20] M.S. Lewicki, T.J. Sejnowski, Learning overcomplete representations, Neural Computation 12 (2000) 337–365.

[21] C.-S. Li, P.S. Yu, V. Castelli, Hierarchyscan: A hierarchical similarity search algorithm for databases of long sequences, in: Proceedings of the International Conference on Data Engineering, New Orleans, LA, February 1996, pp. 546–553.

[22] R.F. Lyon, M. Rehn, S. Bengio, T.C. Walters, G. Chechik, Sound retrieval and ranking using sparse auditory representations, Neural Computation 22 (9) (2010) 2390–2416.

[23] B. Mailhé, R. Gribonval, P. Vandergheynst, F. Bimbot, Fast orthogonal sparse approximation algorithms over local dictionaries, Signal Processing, this issue.

[24] S. Mallat, A Wavelet Tour of Signal Processing: The Sparse Way, third ed., Academic Press, Elsevier, Amsterdam, 2009.

[25] R. Mazhar, P.D. Gader, J.N. Nilson, Matching pursuits dissimilarity measure for shape-based comparison and classification of high-dimensional data, IEEE Transactions on Fuzzy Systems 17 (5) (2009) 1175–1188.

[26] M. Müller, F. Kurth, M. Clausen, Chroma-based statistical audio features for audio matching, in: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, NY, October 2005, pp. 275–278.

[27] Y. Panagakis, C. Kotropoulos, G.R. Arce, Music genre classification via sparse representations of auditory temporal modulations, in: Proceedings of the European Signal Processing Conference Glasgow, Scotland, August 2009, pp. 1–5.

[28] Y. Pati, R. Rezaiifar, P. Krishnaprasad, Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition, in: Proceedings of the Asilomar Conference on Signals, Systems, and Computers, vol. 1, Pacific Grove, CA, November 1993, pp. 40–44.

[29] T.V. Pham, A. Smeulders, Sparse representation for coarse and fine object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 555–567.

[30] D. Rafiei, A. Mendelzon, Efficient retrieval of similar time sequences using DFT, in: Proceedings of the International Conference of Foundations of Data Organization and Algorithms, Kobe, Japan, November 1998, pp. 249–257.

[31] E. Ravelli, G. Richard, L. Daudet, Union of MDCT bases for audio coding, IEEE Transactions on Audio, Speech and Language Processing 16 (8) (2008) 1361–1372.

[32] E. Ravelli, G. Richard, L. Daudet, Audio signal representations for indexing in the transform domain, IEEE Transactions on Audio, Speech and Language Processing 18 (3) (2010) 434–446.

[33] L. Rebollo-Neira, D. Lowe, Optimized orthogonal matching pursuit approach, IEEE Signal Processing Letters 9 (4) (2002) 137–140.

[34] S. Scholler, H. Purwins, Sparse coding for drum sound classification and its use as a similarity measure, in: Proceedings of the International Workshop on Machine Learning Music ACM Multimedia, Firenze, Italy, October 2010.

[35] J. Serrà, E. Gómez, P. Herrera, X. Serra, Chroma binary similarity and local alignment applied to cover song identification, IEEE Transactions on Audio, Speech and Language Processing 16 (August 2008) 1138–1151.

[36] B.L. Sturm, M. Christensen, Cyclic matching pursuit with multiscale time–frequency dictionaries, in: Proceedings of the Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, November 2010.
[37] B.L. Sturm, J.J. Shynk, Sparse approximation and the pursuit of meaningful signal models with interference adaptation, IEEE Transactions on Audio, Speech and Language Processing 18 (3) (2010) 461–472.
[38] B.L. Sturm, J.J. Shynk, A. McLeran, C. Roads, L. Daudet, A comparison of molecular approaches for generating sparse and structured multiresolution representations of audio and music signals, in: Proceedings of Acoustics, Paris, France, June 2008, pp. 5775–5780.
[39] G. Tzanetakis, P. Cook, Musical genre classification of audio signals, IEEE Transactions on Speech, and Audio Processing 10 (5) (2002) 293–302.
[40] K. Umapathy, S. Krishnan, S. Jimaa, Multigroup classification of audio signals using time–frequency parameters, IEEE Transactions on Multimedia 7 (2) (2005) 308–315.
[41] P. Vincent, Y. Bengio, Kernel matching pursuit, Machines Learning. 48 (1) (2002) 165–187.
[42] A. Wang, An industrial strength audio search algorithm, in: Proceedings of the International Society on Music Information Retrieval, Baltimore, Maryland, USA, October 2003, pp. 1–4.
[43] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang, S. Yan, Sparse representation for computer vision and pattern recognition, Proceedings of the IEEE 98 (6) (2009) 1031–1044.
[44] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 210–227.