Audio Inpainting

Amir Adler, Valentin Emiya, Maria G. Jafari, Michael Elad, Rémi Gribonval, Mark D. Plumbley

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Audio Inpainting

Amir Adler — Valentin Emiya — Maria G. Jafari — Michael Elad — Rémi Gribonval — Mark D. Plumbley

N° 7571

March 16, 2011

Domaine Audio, Speech, and Language Processing

ISSN 0249-6399 ISRN INRIA/RR--7571--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
RENNES - BRETAGNE ATLANTIQUE

Audio Inpainting

Amir Adler , Valentin Emiya , Maria G. Jafari , Michael Elad ,
Rémi Gribonval , Mark D. Plumbley

Domaine :
Équipe-Projet Metiss

Rapport de recherche n° 7571 — March 16, 2011 — 24 pages

Abstract:

We propose the Audio Inpainting framework that recovers audio intervals distorted due to impairments such as impulsive noise, clipping, and packet loss. In this framework, the distorted samples are treated as missing, and the signal is decomposed into overlapping time-domain frames. The restoration problem is then formulated as an inverse problem per audio frame. Sparse representation modeling is employed per frame, and each inverse problem is solved using the Orthogonal Matching Pursuit algorithm together with a discrete cosine or a Gabor dictionary. The performance of this algorithm is shown to be comparable or better than state-of-the-art methods when blocks of samples of variable durations are missing. We also demonstrate that the size of the block of missing samples, rather than the overall number of missing samples, is a crucial parameter for high quality signal restoration. We further introduce a constrained Matching Pursuit approach for the special case of audio declipping that exploits the sign pattern of clipped audio samples and their maximal absolute value, as well as allowing the user to specify the maximum amplitude of the signal. This approach is shown to outperforms state-of-the-art and commercially available methods for audio declipping.

Key-words: Inpainting, clipping, sparse representation, matching pursuit.

A. Adler and M. Elad are with the Computer Science Department, The Technion, Haifa 32000, Israel. V. Emiya and R. Gribonval are with INRIA, Centre Inria Rennes - Bretagne Atlantique, 35042 Rennes Cedex, France. M. G. Jafari and M. D. Plumbley are with Queen Mary University of London, Centre for Digital Music, Department of Electronic Engineering, London E1 4NS, U.K., (e-mail: maria.jafari@elec.qmul.ac.uk).

This work has been submitted to IEEE Transactions on Audio Speech and Language Processing. Part of this work has been presented at the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) in 2011 [1].

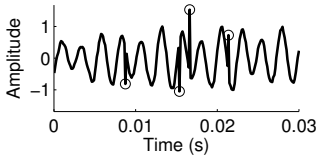
This work was supported by the EU Framework 7 FET-Open project FP7-ICT-225913-SMALL: Sparse Models, Algorithms and Learning for Large-Scale data.

Centre de recherche INRIA Rennes – Bretagne Atlantique
IRISA, Campus universitaire de Beaulieu, 35042 Rennes Cedex
Téléphone : +33 2 99 84 71 00 — Télécopie : +33 2 99 84 71 71

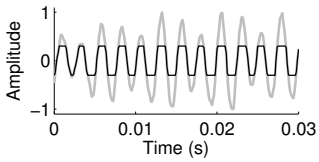
Audio Inpainting

Résumé : Nous introduisons le concept d'Inpainting Audio pour la restauration de portions de données audio distordues par des dégradations tels que les clics, le clipping ou la perte de paquets. Dans ce contexte, les données distordues sont considérées comme manquantes et le signal est décomposé dans le domaine temporel en trames. Le problème de restauration est formulé comme un problème inverse dans chaque trame. Celle-ci est modélisée par une représentation parcimonieuse et le problème inverse est résolu via l'algorithme Orthogonal Matching Pursuit en utilisant un dictionnaire de cosinus discret ou de Gabor. Les performances obtenues sont comparables à l'état de l'art, avec des blocs d'échantillons manquants de durée variable. Nous montrons également que la qualité de la restauration dépend davantage de la taille des blocs d'échantillons manquants que du nombre total d'échantillons manquants. Nous introduisons enfin un algorithme de type Matching Pursuit avec contraintes pour le cas particulier du declipping audio, dans laquelle sont exploitées les propriétés d'amplitude des échantillons saturés: signe, amplitude minimum et maximum. Les performances obtenues sont supérieures à celles de l'état de l'art et à de logiciels commerciaux pour le declipping.

Mots-clés : Inpainting, clipping, représentations parcimonieuses, matching pursuit.



(a) Speech signal corrupted by clicks (circles).



(b) Clipped version (black) of a speech signal (gray).

(c) The image inpainting problem: recovery of locally-hidden pixels.

Figure 1: Examples of restoration problems related to inpainting.

1 Introduction

Speech and music signals are often subject to localized distortions, where the intervals of distorted samples are surrounded by undistorted samples. Examples include impulsive noise or clicks (see Fig. 1a), clipping (see Fig. 1b), CD scratches, packet loss in cordless phones or Voice over IP (VoIP) and more. In such situations, the distorted samples can be treated as missing. A restoration algorithm is employed to reconstruct the missing samples, in a similar way as for image inpainting (see Fig. 1c). However, in the audio field, such problems have been treated separately and depending on the context, they have been referred to as audio interpolation [2–6], extrapolation [3,7,8], imputation [9,10], induction [11], (bandwidth) extension [12–15] or concealment [16,17].

Substantial effort has been focused on the restoration of audio signals corrupted by clicks due to old recordings or scratched CDs (see Fig. 1a). In this problem, intervals of corrupted samples – from 20 μs to 4 ms [4] – occur at random locations. Typical approaches employ autoregressive (AR) modeling [2, 3], or Bayesian estimation to recover the corrupted samples [4]. Other methods utilize neural networks [18] or sinusoidal modeling [5, 8]. A related problem is automatic speech recognition in the presence of isolated noisy samples. This problem is treated in [10] with a compressive sensing approach in the spectrogram image domain, and by solving an l_1 regularized least squares problem.

Another important – though less often addressed – problem is audio clipping [6,7,19], which refers to the truncation of the waveform beyond a threshold when the maximum range in an acquisition system is exceeded (see Fig. 1b). The clipped samples are arranged in groups and their locations are determined by the amplitude of the signal (rather than being randomly spread). The declipping problem is particularly challenging for this reason and as the information carried by the highest-amplitude samples is completely absent.

Long intervals of samples may be lost during transmission over cordless phones or in VoIP systems, where the problem is addressed using packet loss concealment algorithms [16, 17]. Missing intervals lengths are in the range of 5 ms to 60 ms, which are close to the typical duration for the pseudo-stationarity of audio signals. The low latency requirement in the VoIP case results in relatively simple algorithms; however, estimating missing packets in peer-to-peer repositories is a new application where higher quality reconstruction can be expected (as the latency requirement is less stringent).

Finally, the unreliable or missing audio data can be time-frequency regions [5,9,11,14,20], in classification applications like automatic speech recognition [9,20] or source separation with time-frequency localized interference – the phrase "audio inpainting" has been used once in this specific case [11]. Bandwidth extension [12–15] is another important time-frequency-domain application, where high frequency content is estimated from the low frequency content in order to provide high quality audio.

In this paper, we present a unified framework for the restoration of distorted audio data, leveraging the concept of *Image Inpainting* [21–23]. In the proposed framework, termed *Audio Inpainting*, the distorted data is assumed missing and its location is assumed to be known a-priori. We further employ Sparse Representations (SR), which have been demonstrated to faithfully model audio signals [24, 25] and to address the image inpainting framework [22, 26, 27]. The proposed approach is directly based upon those prior works.

The contributions of this paper are four-fold:

a) Audio inpainting is defined as an inverse problem, based upon the concept of image inpainting.

b) A framework for audio inpainting in the time domain is proposed, based on sparse representations. It exploits two possible dictionaries (discrete cosine and Gabor) known to provide accurate sparse models for audio signals.

c) The Orthogonal Matching Pursuit (OMP) algorithm for audio inpainting is adapted, in particular to deal with the properties of the Gabor dictionary.

d) A constrained matching pursuit approach is applied to significantly enhance the performance for audio declipping problems.

This paper is organized as follows. In Section 2, audio inpainting is formalized as an inverse problem. The proposed framework is introduced in Section 3 including the sparse models used for time-domain audio inpainting. The adaptation of the OMP algorithm for audio inpainting in the time domain and for audio declipping is presented in Section 4. Several experiments are proposed in Section 5, while we discuss our findings and draw conclusions in Section 6.

2 Audio Inpainting Problem Statement

We define audio inpainting as a general problem encountered in many applications: one observes a partial set of reliable audio data while the remaining unreliable data is either totally missing or highly degraded; the unreliable data is considered missing and it is estimated from the reliable data portion.

The general formulation of audio inpainting is given in Section 2.1 while several particular time-domain cases are detailed in Sections 2.2 and 2.3.

2.1 Formulation of audio inpainting

We consider a vector $\mathbf{s} \in \mathbb{R}^L$ of audio data and an a-priori known partition $\{I^{\mathrm{m}}, I^{\mathrm{r}}\}$ of the support $I \triangleq \{1, 2, \cdots, L\}$ of $\mathbf{s}$: $I^{\mathrm{m}} \subset I$ and $I^{\mathrm{r}} \triangleq I \backslash I^{\mathrm{m}}$. We assume that the coefficients $\mathbf{s}\,(I^{\mathrm{m}})$ are either missing or masked by a severe distortion. Thus, the observed data $\mathbf{y} \in \mathbb{R}^L$ coincides with $\mathbf{s}$ on I^{r} only. The audio inpainting problem is defined as the recovery of the coefficients $\mathbf{s}\,(I^{\mathrm{m}})$ based on the knowledge of:

1. the reliable data $\mathbf{y}^{\mathrm{r}} \triangleq \mathbf{y}\,(I^{\mathrm{r}}) = \mathbf{s}\,(I^{\mathrm{r}})$,
2. the partition $\{I^{\mathrm{m}}, I^{\mathrm{r}}\}$,
3. additional information about the observed signal,
4. and, optionaly, information about the missing data (see *e.g.* in the case of clipping below).

In matrix form, the reliable data $\mathbf{y}^{\mathrm{r}}$ result from the linear model

$$\mathbf{y}^{\mathrm{r}} = \mathbf{M}^{\mathrm{r}}\mathbf{s}, \tag{1}$$

where $\mathbf{M}^{\mathrm{r}}$ is the so-called measurement matrix obtained from the $L \times L$ identity matrix $\mathbf{I}_L$ by selecting the rows I^{r} associated with the reliable coefficients in $\mathbf{s}$. In a similar way, the missing data to be recovered are $\mathbf{s}\,(I^{\mathrm{m}}) = \mathbf{M}^{\mathrm{m}}\mathbf{s}$, where $\mathbf{M}^{\mathrm{m}}$ consists of the rows I^{m} in $\mathbf{I}_L$.

In the general audio inpainting framework, audio data can be either samples in waveforms or coefficients in transforms like time-frequency representations. The problem formulation above can be used for multi-dimensional signals like multichannel waveforms or time-frequency coefficients, by simply reshaping the signal matrix as a vector $\mathbf{s}$.

In the rest of this paper, we only consider the inpainting of missing samples in a single-channel waveform. The multi-dimensional case is discussed in the conclusion (see Sec. 6).

2.2 Inpainting samples distorted by impulsive noise

In the particular case of a signal corrupted by impulsive noise such as clicks (see Fig. 1a), I^{m} is a set of integers between 1 and L and must be estimated in a preliminary stage. One often considers that the distorted samples are corrupted by a Gaussian noise $\mathbf{n}$ with high variance. Hence, the complete observed signal includes both the reliable samples $\mathbf{y}^{\mathrm{r}}$ and distorted ones $\mathbf{y}^{\mathrm{m}}$:

$$\begin{cases} \mathbf{y}^{\mathrm{r}} &= \mathbf{M}^{\mathrm{r}}\mathbf{s} \\ \mathbf{y}^{\mathrm{m}} &= \mathbf{M}^{\mathrm{m}}\mathbf{s} + \mathbf{n}, \end{cases} \tag{2}$$

where the samples $\mathbf{M}^{\mathrm{m}}\mathbf{s}$ in $\mathbf{y}^{\mathrm{m}}$ are masked by $\mathbf{n}$ so that they are considered as unknown.

2.3 Inpainting intervals of missing samples

In the case where intervals of samples are missing, due to packet loss during transmission or to masking by audible interferences, I^{m} is composed of groups of consecutive integers: the samples $\mathbf{s}\left(I^{\mathrm{m}}\right)$ are totally missing and one only observes $\mathbf{y}^{\mathrm{r}} = \mathbf{M}^{\mathrm{r}}\mathbf{s}$.

In the case of clipped signals, the samples to be estimated are also arranged in intervals of consecutive samples, as depicted in Fig. 1b. Their locations depend on the amplitude of the signal, such that

$$I^{\mathrm{m}} \triangleq \left\{ n \,|1 \leq n \leq L, |\mathbf{s}\left(n\right)| \geq \theta_{\mathrm{clip}} \right\}, \tag{3}$$

where θ_{clip} is the clipping level. One observes both the un-clipped, reliable samples $\mathbf{y}^{\mathrm{r}}$ and the clipped, masked samples $\mathbf{y}^{\mathrm{m}}$

$$\begin{cases} \mathbf{y}^{\mathrm{r}} & = \mathbf{M}^{\mathrm{r}}\mathbf{y} = \mathbf{M}^{\mathrm{r}}\mathbf{s} \\ \mathbf{y}^{\mathrm{m}} & = \mathbf{M}^{\mathrm{m}}\mathbf{y} = \mathbf{M}^{\mathrm{m}} \operatorname{sign}\left(\mathbf{s}\right) \theta_{\mathrm{clip}}, \end{cases} \tag{4}$$

where $\operatorname{sign}\left(\cdot\right)$ is the element-wise sign function. As presented in the next sections, the information provided by $\mathbf{y}^{\mathrm{m}}$, even though very crude – a sign (per sample) and the clipping level –, still substantially enhances the estimation performance.

3 Time-domain framework and models

The proposed framework focuses on time-domain audio inpainting. It relies on a frame-based processing, as detailed in Section 3.1 and on the sparse representations modeling of audio signals, as presented in Section 3.2. Two dictionaries used in this modeling are introduced in Section 3.3.

3.1 Frame-based processing and reconstruction

As in many audio processing tasks, the signal is locally processed:

- by segmenting it into frames,
- by independently inpainting each frame,
- and by synthesizing the full restored signal using the overlap-add (OLA) method [28].

We decompose the signal into overlapping frames indexed by i, starting at time t_i and weighted by an analysis window $\mathbf{w}_a$ with length N. By straightforwardly adapting to the local frames the problem statement defined for the full signal in Section 2, the reliable samples in frame i can be written as

$$\mathbf{y}_i^{\mathrm{r}} = \mathbf{M}_i^{\mathrm{r}}\mathbf{s}_i \tag{5}$$

where $\mathbf{M}_i^{\mathrm{r}}$ is the measurement matrix of the i-th frame obtained from $\mathbf{M}^{\mathrm{r}}$ and $\mathbf{s}_i(t) \triangleq \mathbf{s}(t+t_i)\,\mathbf{w}_a(t)$ is the windowed frame defined for $0 \leq t \leq N-1$. We also define the supports I_i^{r} and I_i^{m} of the reliable samples and of the missing or masked samples, respectively. Once the estimation $\widehat{\mathbf{s}}_i$ of $\mathbf{s}_i$ by some inpainting algorithm is achieved, the reconstruction of the full signal is obtained as

$$\widehat{\mathbf{s}}(t) \triangleq \sum_i \mathbf{w}_s(t-t_i)\,\widehat{\mathbf{s}}_i(t-t_i) \tag{6}$$

where $\mathbf{w}_s$ is the synthesis window such that $\sum_i \mathbf{w}_s(t-t_i)\,\mathbf{w}_a(t-t_i) = 1, \forall t$. In the proposed approaches, we utilized 64ms-frames with 75% overlap, a rectangular window for $\mathbf{w}_a$ and a sine window for $\mathbf{w}_s$.

3.2 Sparse Representations modeling of audio frames

In the Sparse Representations (SR) modeling framework [23], it is assumed that each frame is well approximated by a sparse linear combination of the columns of a (possibly overcomplete) dictionary:

$$\mathbf{s}_i \approx \mathbf{D}\mathbf{x}_i, \tag{7}$$

where $\mathbf{D} \in \mathbb{R}^{N \times K_{\mathbf{D}}}$ is the dictionary, $N \leq K_{\mathbf{D}}$ and $\mathbf{x}_i \in \mathbb{R}^{K_{\mathbf{D}} \times 1}$ is the representation vector of the i-th frame. $\mathbf{x}_i$ is assumed to be sparse, *i.e.* to have few non-zero coefficients compared to N. As a consequence, we can also utilize the SR model for the observed reliable samples in eachframe

$$\mathbf{y}_i^{\mathrm{r}} \triangleq \mathbf{M}_i^{\mathrm{r}}\mathbf{s}_i \approx \mathbf{M}_i^{\mathrm{r}}\mathbf{D}\mathbf{x}_i. \tag{8}$$

We propose to recover the unknown samples $\mathbf{s}_i(I_i^{\mathrm{m}})$ by estimating as $\hat{\mathbf{x}}_i$ the (sparse) representation vector of each frame, given only the clean observed samples (8) and limited side information (for the clipping case)

$$\widehat{\mathbf{s}}_i(I_i^{\mathrm{m}}) = \mathbf{M}_i^{\mathrm{m}}\mathbf{D}\hat{\mathbf{x}}_i. \tag{9}$$

This formulation including the notion of sparsity was first introduced for image inpainting [22] with a global treatment with global transforms. Then, efforts were dedicated to work on local patches – similar to audio frames – and to introduce a learned dictionary to improve the inpainting results [26]; they have been improved [27] by modeling better the problem and by learning the dictionary directly from the corrupted image.

3.3 Dictionaries

We propose two options to choose a dictionary $\mathbf{D}$ in which audio signals are sparse: the Discrete Cosine Transform dictionary, and a Gabor dictionary. Both are widely used for sparse models of audio signals [24, 25, 29]. Other fixed dictionaries such as multiscale DCT [30], or learned dictionary [26] specific to particular inpainting tasks may also be interesting options.

3.3.1 Discrete Cosine Transform (DCT) dictionary

The first option consists in a windowed Discrete Cosine Transform (DCT) overcomplete dictionary $\mathbf{D}^c = \left[\mathbf{d}^c_0, \ldots, \mathbf{d}^c_{K_c-1}\right]$, atom j being defined for $0 \leq j \leq K_c - 1$ and $0 \leq t \leq N-1$ as

$$\mathbf{d}^c_j(t) \triangleq \mathbf{w}_d(t) \cos\left(\frac{\pi}{K_c}\left(t+\frac{1}{2}\right)\left(j+\frac{1}{2}\right)\right) \tag{10}$$

where K_c is the size of the DCT dictionary – *i.e.* the number of discrete frequencies – and $\mathbf{w}_d$ is a weighting window set by the user. This choice is motivated by the wide use of windowed DCT atoms for sparse representation of audio signals [25]. However, the zero phase of $\mathbf{D}^c$ atoms is not adapted to audio signals that are made up with sinusoidal components with initial phase distributed between 0 and 2π. As a consequence, the DCT model acts as a basis rather than as a synthesis model and the signals are not really sparse in $\mathbf{D}^c$.

3.3.2 Gabor dictionary

The second option aims at sparsely modeling arbitrary-phase sinusoidal components by using a Gabor dictionary $\mathbf{D}^g = \left\{\mathbf{d}^g_{(j,\varphi)}\right\}_{(j,\varphi)\in\Gamma}$ in which the atoms are index by a continuous set $\Gamma \triangleq [\![0, K_g - 1]\!] \times [0, 2\pi[$ and are defined as

$$\mathbf{d}^g_{j,\varphi} \triangleq \mathbf{w}_d(t) \cos\left(\frac{\pi}{K_g}\left(t+\frac{1}{2}\right)\left(j+\frac{1}{2}\right)+\varphi\right), \tag{11}$$

where K_g is the size of the Gabor dictionary.

Note that in the current case of a continuously-indexed dictionary, eq. (7), (8) and (9) are still valid if we define

$$\mathbf{D}^g\mathbf{x}_i = \sum_{\substack{(j,\varphi)\in\Gamma \\ \mathbf{x}_i(j,\varphi)\neq 0}} \mathbf{d}^g_{j,\varphi}\mathbf{x}_i(j,\varphi) \tag{12}$$

where $\mathbf{x}_i = \{\mathbf{x}_i(j,\varphi)\}_{(j,\varphi)\in\Gamma}$. Indeed, eq. (12) is a finite sum since only a few coefficients in the sparse representation vector $\mathbf{x}_i$ are non-zero. The algorithmic aspects of this decomposition will be addressed in Sections 4.2 and 4.3.

4 Audio inpainting algorithms based on Orthogonal Matching Pursuit

For a given dictionary $\mathbf{D}$, we use the Orthogonal Matching Pursuit algorithm to perform the inpainting of an audio frame, as presented in Section 4.1. Some dictionary-dependent algorithmic stages are then detailed in Section 4.2 and 4.3. An extension of the algorithm specific to declipping is finally detailled in Section 4.4.

Table 1: OMP Inpainting Algorithm

$\mathbf{y}_i^{\mathrm{r}}$, $\mathbf{M}_i^{\mathrm{r}}$, $\mathbf{D} = \{\mathbf{d}_j\}_{j\in\Gamma}$, K^{OMP}, $\epsilon_i^{\mathrm{OMP}}$
Initialization: • Dictionary $\widetilde{\mathbf{D}} = \left\{\widetilde{\mathbf{d}}_j\right\}_{j\in\Gamma} = \mathbf{M}_i^{\mathrm{r}} \times \mathbf{D} \times \mathbf{W}$, where $\mathbf{W}_{jj'} = 0$ for $j \neq j'$ and $\mathbf{W}_{jj} = \|\mathbf{M}_i^{\mathrm{r}}\mathbf{d}_j\|_2^{-1}$. • Iteration counter $k = 0$ • Support set $\Omega_0 = \emptyset$ • Residual $\mathbf{r}_0 = \mathbf{y}_i^{\mathrm{r}}$
Sparse support selection and coefficients estimation: Repeat until $k = K^{\mathrm{OMP}}$ or $\|\mathbf{r}_k\|_2^2 < \epsilon_i^{\mathrm{OMP}}$ • Increment iteration counter $k = k + 1$ • Select atom: find $j = \arg\max_{j\in\Gamma} \lvert < \mathbf{r}_{k-1}, \widetilde{\mathbf{d}_j} > \rvert$ (14) • Update Support $\Omega_k = \Omega_{k-1} \cup j$ • Update current solution $\mathbf{x}_k = \arg\min_{\mathbf{u}} \|\mathbf{y}_i^{\mathrm{r}} - \widetilde{\mathbf{D}}_{\Omega_k}\mathbf{u}\|_2$ (15) • Update Residual $\mathbf{r}_k = \mathbf{y}_i^{\mathrm{r}} - \widetilde{\mathbf{D}}_{\Omega_k}\mathbf{x}_k$
Output: $\hat{\mathbf{x}}_i = \mathbf{W}\mathbf{x}_k$

4.1 Orthogonal Matching Pursuit (OMP) algorithm for inpainting

The approach emerges from the following optimization problem

$$\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \|\mathbf{y}_i^{\mathrm{r}} - \mathbf{M}_i^{\mathrm{r}}\mathbf{D}\mathbf{x}\|_2^2 \leq \epsilon_i. \tag{13}$$

for a given approximation error threshold ϵ_i.

The l_0 pseudo-norm $\|\mathbf{x}\|_0$ counts the non-zeros components of the vector $\mathbf{x}$, leading to an NP-hard problem [31, 32]. Therefore, a direct solution of (13) is infeasible. An approximate solution is given by applying the Orthogonal Matching Pursuit (OMP) algorithm [24, 33], which successively approximates the sparsest solution. The inpainting OMP algorithm [23], detailed in Table 1, is a slightly modified version of the classical OMP algorithm in the sense that all dictionary columns $\widetilde{\mathbf{d}}_j$ are internally normalized to unit norm, using diagonal matrix $\mathbf{W}$, due to the availability of only the clean samples. The algorithm stops iterating as soon as either the residual energy drops below the threshold $\epsilon_i^{\mathrm{OMP}}$ or the maximum sparsity level K^{OMP} is exceeded.

4.2 Atom selection

When using the DCT dictionary, the algorithmic stage for the atom selection (14) at each iteration is well known: it consists of explicitly computing the correlation mentioned in eq. (14) of Table 1, or of using a fast transform.

However, the atom selection (14) needs explaining in the case of the Gabor dictionary in order to deal with the continuous indexing. Without any approximation, the decomposition with continuously-indexed atoms $\mathbf{d}^g$ can be expressed using pairs of atoms in a discrete dictionary with K_g frequency bins. Pairs of atoms can be either conjugate complex exponentials [24,34], or pairs of cosine and sine at the same frequency and with a zero phase [29]. In order to use this latter option, we introduce sine atoms $\mathbf{d}_j^s$ as

$$\mathbf{d}_j^s(t) \triangleq \mathbf{w}_d(t) \sin\left(\frac{\pi}{K_g}\left(t+\frac{1}{2}\right)\left(j+\frac{1}{2}\right)\right) \tag{16}$$

and we define the unit-norm version $\widetilde{\mathbf{d}}_j^c$ and $\widetilde{\mathbf{d}}_j^s$ of the atoms $\mathbf{d}_j^c$ and $\mathbf{d}_j^s$, respectively, as described in Table 1.

At each iteration k, selecting the best correlated Gabor atom $\mathbf{d}_j^g$ (eq. (14)) is then equivalent to picking the pair $\left(\widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s\right)$ such that

$$j = \underset{j\in[\![1,K_{\mathbf{D}}/2]\!]}{\arg\min} \left\| \mathbf{r}_{k-1} - \widetilde{\mathbf{d}}_j^c \widehat{x}_j^c - \widetilde{\mathbf{d}}_j^s \widehat{x}_j^s \right\|_2^2 \tag{17}$$

where

$$\begin{cases} \widehat{x}_j^c &= \dfrac{\left\langle \widetilde{\mathbf{d}}_j^c, \mathbf{r}_{k-1}\right\rangle - \left\langle \widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s\right\rangle \left\langle \widetilde{\mathbf{d}}_j^s, \mathbf{r}_{k-1}\right\rangle}{1-\left\langle \widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s\right\rangle^2} \\ \widehat{x}_j^s &= \dfrac{\left\langle \widetilde{\mathbf{d}}_j^s, \mathbf{r}_{k-1}\right\rangle - \left\langle \widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s\right\rangle \left\langle \widetilde{\mathbf{d}}_j^c, \mathbf{r}_{k-1}\right\rangle}{1-\left\langle \widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s\right\rangle^2} \end{cases}. \tag{18}$$

This particular selection stage has been proposed in [34, Appendix II] for conjugate Gabor chirp atoms and the use of blocks of coherent atoms in MP and OMP has been further studied in [35]. In the restricted case where atoms in a candidate pair are uncorrelated (*i.e.* $\left\langle \widetilde{\mathbf{d}}_j^c, \widetilde{\mathbf{d}}_j^s \right\rangle = 0$), eq. (17) can be simplified as $j = \arg\max_{j\in[\![1,K_{\mathbf{D}}/2]\!]} \left\langle \widetilde{\mathbf{d}}_j^c, \mathbf{r}_{k-1} \right\rangle^2 + \left\langle \widetilde{\mathbf{d}}_j^s, \mathbf{r}_{k-1} \right\rangle^2$ and the resulting algorithm is equivalent to the existing Modified Matching Pursuit [36] and Block OMP algorithms [37].

4.3 Solution update

When using the DCT dictionary, the solution update (15) performed at each iteration usually consists of a least-square projection.

In the case of the Gabor dictionary, once the best atom has been added to the set of atoms selected in previous iterations, the update of the current solution (15) can be performed by a least-square projection using the selected Gabor atoms $\left\{\mathbf{d}_{j,\varphi_j}^g\right\}_{j\in\Omega_k}$, their phases φ_j being fixed in the atom-selection stage.

However, this update can be improved by using the equivalent cosine and sine atoms $\left\{\mathbf{d}_j^c, \mathbf{d}_j^s\right\}_{j\in\Omega_k}$ in the least-square projection: not only amplitudes but also phases are thus updated at each iteration, leading to a better approximation of the signal. As far as we know, such an implementation of OMP with a Gabor dictionary has never been proposed before.

4.4 Algorithmic enhancements for inpainting clipped signals

4.4.1 The 'min' declipping constraint

Inpainting clipped signals can be performed with the algorithm presented in Section 4.1, by treating the clipped samples as completely unknown. However, extra information inherent to this problem can be integrated as additional constraints into equations (13). Constrained optimization approaches were also utilized in the case of l_1-minimization for image desaturation [38] and of audio declipping based on a band-limited assumption [7].

Let θ_{clip} be the clipping level (which can be easily estimated as the maximum absolute value among the observed samples) and $\mathbf{M}_i^{\text{m}+}$ (resp. $\mathbf{M}_i^{\text{m}-}$) be the matrix such that $\mathbf{M}_i^{\text{m}+}\mathbf{s}_i$ (resp. $\mathbf{M}_i^{\text{m}-}\mathbf{s}_i$) is the vector of positive (resp. negative) clipped samples. The matrices $\mathbf{M}_i^{\text{m}+}$ and $\mathbf{M}_i^{\text{m}-}$ are known from the location and the sign of the clipped samples. The missing samples should satisfy the 'min' constraints

$$\mathbf{M}_i^{\text{m}+}\mathbf{s}_i \geq \theta_{\text{clip}} \text{ and } \mathbf{M}_i^{\text{m}-}\mathbf{s}_i \leq -\theta_{\text{clip}}. \tag{19}$$

4.4.2 The 'max' declipping constraint

The set of 'min' constraints can be further augmented by 'max' constraints, introducing an upper limit on the absolute value of the recovered samples $\theta_{\max}$, as follows

$$\mathbf{M}_i^{\text{m}+}\mathbf{s}_i \leq \theta_{\max} \text{ and } \mathbf{M}_i^{\text{m}-}\mathbf{s}_i \geq -\theta_{\max}. \tag{20}$$

The upper limit $\theta_{\max}$ is an optional parameter that cannot be estimated automatically in a straightforward way but may be adjusted manually by the user.

4.4.3 The 'minmax' constrained SR problem

Using both sets of constraints, the 'minmax' declipping version of the l_0-norm minimization problem (13) is given by

$$\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \begin{cases} \|\mathbf{y}_i^{\text{r}} - \mathbf{M}_i^{\text{r}}\mathbf{D}\mathbf{x}\|_2^2 & \leq \epsilon_i \\ \theta_{\max} \geq \mathbf{M}_i^{\text{m}+}\mathbf{D}\mathbf{x} & \geq \theta_{\text{clip}} \\ -\theta_{\max} \leq \mathbf{M}_i^{\text{m}-}\mathbf{D}\mathbf{x} & \leq -\theta_{\text{clip}} \end{cases} \tag{21}$$

where $\theta_{\max}$ can be set to $+\infty$ if one does not want to use the 'max' constraint.

Table 2: Summary of the proposed algorithms: each row indicates the algorithm usage (general inpainting or declipping), depending on possible additional constraints, while dictionaries vary accross columns. Algorithm nomenclature appears within quotes in each cell.

Additional specification	DCT	Gabor
Inpainting	'OMP-C' [Table 1]	'OMP-G' [Table 1]
Min-constraint declipping	'OMP-C-min' [Table 1 + eq. (22)]	'OMP-G-min' [Table 1 + eq. (22)]
Minmax-constraint declipping	'OMP-C-minmax' [Table 1 + eq. (23)]	'OMP-G-minmax' [Table 1 + eq. (23)]

4.4.4 OMP declipping algorithm

We propose approximate solutions by incorporating the constraints (19) and (20) into the final solution update stage of the OMP Inpainting algorithm. In other words, the OMP Inpainting algorithm presented in Table 1 is applied, in order to *select* the sparse support. Once the support Ω_k is selected, the sparse representation coefficients are *re-estimated* by solving the following constrained optimization problem:

$$\mathbf{x}_k = \arg\min_{\mathbf{u}} \|\mathbf{y}_i^{\mathrm{r}} - \widetilde{\mathbf{D}}_{\Omega_k}\mathbf{u}\|_2 \text{ s.t. } \begin{cases} \mathbf{M}_i^{\mathrm{m}+}\mathbf{D}\mathbf{W}\mathbf{u} \geq \hat{\theta}_{\mathrm{clip}} \\ \mathbf{M}_i^{\mathrm{m}-}\mathbf{D}\mathbf{W}\mathbf{u} \leq -\hat{\theta}_{\mathrm{clip}} \end{cases} \tag{22}$$

in the case of the 'min' constraint, or

$$\begin{aligned} \mathbf{x}_k &= \arg\min_{\mathbf{u}} \|\mathbf{y}_i^{\mathrm{r}} - \widetilde{\mathbf{D}}_{\Omega_k}\mathbf{u}\|_2 \\ &\text{s.t. } \begin{cases} \hat{\theta}_{\max} \geq \mathbf{M}_i^{\mathrm{m}+}\mathbf{D}\mathbf{W}\mathbf{u} \geq \hat{\theta}_{\mathrm{clip}} \\ -\hat{\theta}_{\max} \leq \mathbf{M}_i^{\mathrm{m}-}\mathbf{D}\mathbf{W}\mathbf{u} \leq -\hat{\theta}_{\mathrm{clip}} \end{cases} \end{aligned} \tag{23}$$

for the case of the 'minmax' constraint. The constraints are linear, thus standard convex optimization solvers can be employed.

In theory, the solution of the constrained problem may not exist. We observed that this occurs very seldom in practice. Whenever no solution exists, the frame is restored using the unconstrained minimization $\mathbf{x}_k = \arg\min_{\mathbf{u}} \|\mathbf{y}_i^{\mathrm{r}} - \widetilde{\mathbf{D}}_{\Omega_k}\mathbf{u}\|_2$.

5 Experimental Results

A summary of all versions of the algorithm presented in this paper is given in Table 2. This section reports the major trends through different experiments. The performance measures are introduced in Section 5.1. The test material and parameter settings are presented in Sections 5.2 and 5.3. The global performance of all the proposed inpainting algorithms and a more detailed inpainting experiment are presented in Section 5.4. Section 5.5 finally focuses on the case of clipping.

5.1 Performance measures

The performance can be assessed by the signal-to-noise ratio (SNR) computed on the full signals, defined by

$$\mathrm{SNR}_{\mathrm{full}}\left(\mathbf{s},\widehat{\mathbf{s}}\right) \triangleq 10\log\frac{\left\|\mathbf{s}\right\|_2^2}{\left\|\mathbf{s}-\widehat{\mathbf{s}}\right\|_2^2}. \tag{24}$$

While $\mathrm{SNR}_{\mathrm{full}}$ gives an overview of the global quality of the restored signal, it can be decomposed as

$$\mathrm{SNR}_{\mathrm{full}}\left(\mathbf{s},\widehat{\mathbf{s}}\right) = \mathrm{SNR}_{m}\left(\mathbf{s},\widehat{\mathbf{s}}\right) + 10\log\frac{\left\|\mathbf{s}\right\|_2^2}{\left\|\mathbf{s}\left(I^{\mathrm{m}}\right)\right\|_2^2} \tag{25}$$

where

$$\mathrm{SNR}_{m}\left(\mathbf{s},\widehat{\mathbf{s}}\right) \triangleq 10\log\frac{\left\|\mathbf{s}\left(I^{\mathrm{m}}\right)\right\|_2^2}{\left\|\mathbf{s}\left(I^{\mathrm{m}}\right)-\widehat{\mathbf{s}}\left(I^{\mathrm{m}}\right)\right\|_2^2}. \tag{26}$$

SNR_m reflects the reconstruction performance per estimated sample and differs from $\mathrm{SNR}_{\mathrm{full}}$ by an offset that does not depend on the inpainting algorithm. Indeed, the second term in Eq. (25) is a bias that reflects the degradation rate only. Thus, SNR_m will be preferred to show some detailed performance, without the influence of this bias, while $\mathrm{SNR}_{\mathrm{full}}$ will be used to assess the global restoration quality.

Note that in the context of a perceptually-motivated evaluation of the results, SNR measures may be replaced by scores from listening tests or by objective measures. To the authors' knowledge, existing subjective test protocols and objective measures for audio quality assessment are not dedicated to the evaluation of audio inpainting. Indeed, they generally apply to signals that suffer from global degradation rather than local ones, in applications such as coding [39, 40] or source separation [41]. Thus, working on the evaluation of audio inpainting is an important future direction to consider.

5.2 The collection of tested signals

The experiments are conducted using three datasets:

- Music@16kHz: a set of music signals sampled at 16 kHz, this sampling rate being a good trade-off between audio quality and computational requirements.
- Speech@16kHz: a set of speech signals sampled at 16 kHz, *i.e.* high quality speech for which results can be compared to the previous case of music signals.
- Speech@8kHz: a set of speech signals sampled at 8 kHz, representing phone-quality speech; this dataset was obtained by downsampling the previous 16 kHz speech dataset.

Each dataset is composed by ten 5-seconds signals from the 2008's Signal Separation Evaluation Campaign [42] and are freely available online[1]. They include a large diversity of audio mixtures and isolated sources: male and female speech from different speakers, singing voice, pitched and percussive musical instruments.

In order to have comparable degradations among all signals in the clipping experiments (Section 5.5), each original signal is normalized so that the maximum amplitude is 1.

5.3 Parameter settings

A specific training dataset was used to tune the parameters of the inpainting algorithms manually and without fine adjustment. The values of the tuned parameters are shown in Table 3.

Table 3: Parameter settings

Parameter	Value
Frame length	64 ms (*i.e.* $N \triangleq 512$ at 8 kHz, $N \triangleq 1024$ at 16 kHz)
Frame Overlap	75%
Analysis window $\mathbf{w}_a$	rectangular
Synthesis window $\mathbf{w}_s$	sine
Dictionary size	$K_c = 2N$ (DCT), $K_g = N$ (Gabor)
Atom weighting window $\mathbf{w}_d$	rectangular ($\mathbf{w}_d = \mathbf{w}_a$)
$\epsilon_i^{\mathrm{OMP}}$	$\epsilon \times \#I_i^{\mathrm{r}}$ where $\epsilon \triangleq 10^{-6}$ is a fixed parameter and $\#I_i^{\mathrm{r}}$ is the number of reliable samples in the ith frame
K^{OMP}	$\frac{N}{4}$
$\hat{\theta}_{\mathrm{clip}}$	$\|\mathbf{y}\|_\infty$
$\hat{\theta}_{\mathrm{max}}$	$4\hat{\theta}_{\mathrm{clip}}$

5.4 Inpainting experiments

5.4.1 Global effect of the duration of missing intervals

The inpainting performance of the proposed algorithms was evaluated for variable durations of missing intervals of samples. Each experiment tested the performance with the entire collection of signals, for a fixed missing interval duration that repeated periodically every 100 ms. The missing intervals durations were in the range of a fraction of 1ms (corresponding to impulsive noise or clicks distortions) and up to 10 ms (corresponding to packet loss scenarios).

For comparison, we used the method by Janssen [2] based on linear prediction and a reconstruction method based on spline interpolation – the Matlab 'interp1'

[1] http://www.irisa.fr/metiss/vemiya/inpainting/ (this url may be changed to a more stable one by the submission of the final version of this paper)

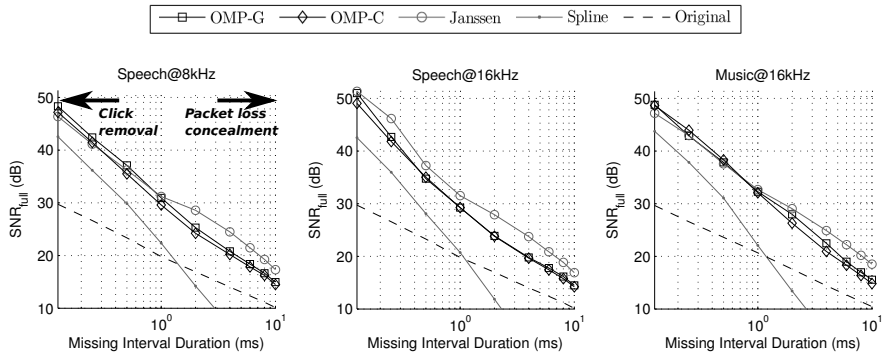


Figure 2: Performance of inpainting algorithms as a function of the duration of missing intervals for each dataset (subfigures). The missing intervals were generated periodically every 100ms (a total of 50 equal duration missing intervals per signal).

function. These methods are representatives of the two main families of state-of-the-art methods for interpolation of audio data and are able to handle multiple blocks of consecutive missing samples. In Janssen's method, the autoregressive model order is set to $3N_{\text{miss}} + 2$, where N_{miss} is the number of missing samples in the current frame, as recommended by the authors.

The results are presented in Fig. 2. On the average, the OMP algorithm with the Gabor dictionary provides an advantage of 1-2dB compared to the OMP with DCT dictionary. For Music@16Khz and Speech@8Khz these algorithms also outperform Janssen's approach for short missing intervals of durations up to 1 ms. For durations above 1ms Janssen's approach provides an advantage of 1-3dB. For Speech@16Khz, Janssen's method performs better than the proposed ones, linear prediction being particularly well-adapted for speech. However, using more information with the proposed method can enhance the performance, as will be shown in the declipping experiment in Section 5.5. The spline interpolation approach provides substantially worse results for all cases.

5.4.2 Fine effect of the 'topology' of the missing samples

The recovery or approximation performance of sparse approaches is often assessed as a function of the sparsity degree and the number of observations in the case of a random measurement matrix [43]. However, the latter assumption, recently highlighted in the compressed sensing framework, does not hold in many audio inpainting applications: as introduced in the previous experiment, one must deal with blocks of consecutive missing samples. In this experiment, we question this assumption and assess empirical performance as a function of the randomness and the consecutiveness of the location of the missing samples. The maximum randomness is achieved when the missing samples are isolated and distributed according to e.g. a uniform law. Conversely, when they are grouped, the missing samples in a given block are not randomly located, even if the blocks themselves may be randomly located. Hence the question: for a fixed number of missing samples, to which extent is the inpainting of few large

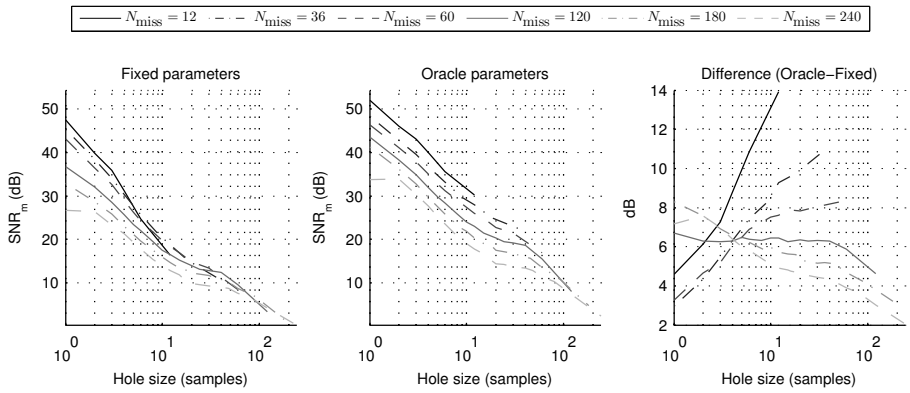


Figure 3: Performance of the OMP-G algorithm, on the Speech@8kHz dataset, as a function of the hole size, for different values of the number N_{miss} of missing samples in a frame: estimation by the proposed algorithm with fixed parameters (left), ideal estimation with the best stopping parameters K^{OMP} and ϵ selected for each observed frame (center) and performance difference (right). The frame length is 512 samples and the hole size ranges from 1 sample (*i.e.* 0.12 ms, 0.2% of the frame) to 240 samples (*i.e.* 30 ms, 46.8% of the frame).

blocks a more difficult problem than the inpainting of many small blocks (or isolated samples)?

The experimental protocol consists in the following steps:

- Choose a set of frames[2]
- Fix the number of missing samples N_{miss};
- For each $(a, b) \in \mathbb{N}^2$ such that $a \times b = N_{\text{miss}}$;
  - For each frame in the set,
    - Randomly generate a holes with length b;
    - Recover the samples inside the holes from the samples outside the holes;
    - Compute SNR_m;
  - Average the SNR_m values w.r.t. all frames.

We use the OMP-G algorithm to recover the samples. The set of values for the number of missing samples N_{miss} is $\{12, 36, 60, 120, 180, 240\}$, which allow a large number of factorizations of the form $a \times b = N_{\text{miss}}$, $(a, b) \in \mathbb{N}^2$. For each test point (N_{miss}, a, b), one thousand 64 ms frames from the 8 kHz speech dataset are processed.

Results are presented in the left plot of Fig. 3. When the hole size is 1 – *i.e.* samples are randomly and uniformly distributed –, the recovery performance is

[2] The frames are randomly chosen in the datasets, only ensuring that the energy in the selected frames is high enough – *i.e.* down to -10 dB below the frame with maximum energy – to avoid silences.

very high with SNR_m values above 35 dB, including the case where the number of missing samples is high (*e.g.* $N_{\mathrm{miss}} = 120$). When holes get larger, the performance significantly decreases: thus, inpainting a single 12-length hole happens to be a much more difficult problem than inpainting a frame with 120 isolated missing samples. Yet, a positive performance is still obtained for the largest holes (*e.g.* $\mathrm{SNR}_m \approx 5dB$ at $N_{\mathrm{miss}} = 100$).

The sensitivity of OMP-G to the stopping criteria was measured thanks to an oracle algorithm. It consists in applying the OMP-G algorithm with different values of $\left(K^{\mathrm{OMP}}, \epsilon\right)$, and in selecting the set of parameters that gives the best performance for each frame independently. The tested parameters were $\left(K^{\mathrm{OMP}}, \epsilon\right) \in \left\{\frac{N}{2^{1.5}}, \frac{N}{2^2}, \ldots, \frac{N}{2^{4.5}}\right\} \times \left\{10^{-10}, 10^{-9}, \ldots, 10^{-1}\right\}$. Results are presented in the center plot of Fig. 3 and the difference between the oracle and blind systems is shown in the right plot. One can see that fixing parameters is a convenient, simple approximation that leads to suboptimal but satisfying performance compared to the oracle case. However, adapting the parameters to the frame to process may be worth studying: the difference between oracle and blind performance ranges from 4 to 10 dB in most of cases, showing a significant potential for improvement.

5.5 Declipping experiment

Clipping restoration is illustrated in Fig. 4 when the clipping level is 0.2. Here, the OMP-C-minmax algorithm is applied to an example of music signal, where one can observe that the reconstructed samples are close to the original signal.

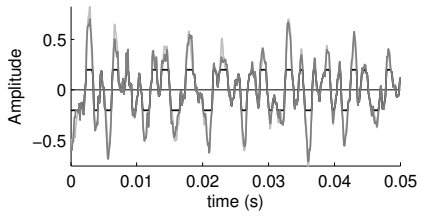


Figure 4: Restoration of a music signal: original (light gray), clipped (black), estimate by the OMP-C-minmax algorithm (dark gray).

In a larger experiment, some of the proposed methods for restoring clipped signals are tested on the 3 datasets Speech@8kHz, Speech@16kHz and Music@16kHz. Each sound is artificially clipped with successive clipping levels, from 0.2 up to 0.9 with a 0.1-step. For this experiment, we selected the OMP-C-minmax, OMP-G, OMP-G-min and OMP-G-minmax algorithms after testing all the algorithms, since the results provide the most interesting conclusions (see below).

The performance of those algorithms are reported in Fig. 5, and show that:

- The use of the 'min' or 'minmax' declipping constraint results in a large improvement of the SNR, on the average by 3 dB for OMP-G. A similar improvement has been obtained in the case of OMP-C. As in previous experiments, we see that methods based on SR, if efficient under random-measurement conditions [23], cannot straightforwardly recover partially-

sampled signals when groups of missing samples are involved. But they are flexible enough to integrate additional constraints that leads to high performance.

- The minmax-constraint OMP-G-minmax algorithm reaches better results than the min-constraint OMP-G-min algorithm when the clipping level is 0.2. This corresponds to the range where the approximate value $\hat{\theta}_{\max}$ is close to the actual maximum value as well as to the most degraded signals. A close analysis of the individual restored sounds reveals that large spikes are avoided thanks to the maximum value constraint. In a practical application, the maximum value $\hat{\theta}_{\max}$ should be adjusted by the user until the best audio quality is achieved.

- The comparison between OMP-C-minmax and OMP-G-minmax shows that the initial-phase modeling by the Gabor dictionary significantly improves the performance, as already observed in Section 5.4.1.

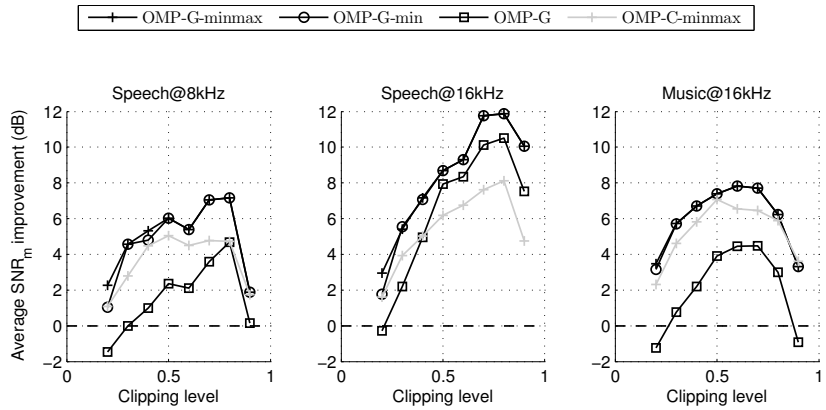
Performance comparison is obtained using two concurrent methods: the *ClipFix* Audacity plug-in based on cubic interpolation, the Cute Studio Declip commercial software[3] and Janssen's method [2] based on linear prediction. The OMP-G-minmax algorithm is compared against these methods and results are shown in Fig. 6. On the average, OMP-G-minmax outperform Janssen's method by 2.8 dB for the Speech@8kHz dataset; by 0.5 dB for the Speech@16kHz dataset; and by 3 dB for the Music@16kHz dataset. Lower performance is obtained from the Cute Studio Declip software, for which the underlying restoration technique is unknown. The *ClipFix* plug-in reaches poor results, below all the reported ones.

6 Conclusions

In this paper, we have presented the Audio Inpainting framework as the general problem of restoring distorted or missing audio data based on the available reliable data. We have defined Audio Inpainting as an inverse problem, and following from image inpainting approaches, we have proposed to use sparse representation methods to restore in the time domain the audio samples that are distorted or missing.

Using a frame-based processing of the audio signal, we have adapted the Orthogonal Matching Pursuit algorithm to address the Audio Inpainting problem, with either a discrete cosine or Gabor dictionary. The performance of this algorithm has been shown to be comparable to or better than state-of-the-art methods when blocks of samples of variable durations were missing, and OMP with the Gabor dictionary has been found to give better results than OMP with DCT dictionary. Moreover, it has been shown that the size of the block of missing samples is more crucial for good signal restoration than the overall number of missing samples to estimate. For the special case of audio declipping, a constrained matching pursuit approach has been applied, that takes into account a priori and user-specified knowledge about the amplitude of the restored signal. This approach has been shown to significantly enhance the performance of the algorithm, which also outperforms state-of-the-art and commercially available

[3] http://www.cutestudio.net/data/products/audio/seedeclip/



Clipping level	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Speech@8kHz	6.2dB	8.2dB	10.1dB	11.9dB	13.7dB	16.5dB	19.2dB	24.4dB
Speech@16kHz	6.3dB	8.4dB	10.3dB	12.2dB	14.2dB	16.8dB	19.8dB	24.8dB
Music@16kHz	6.4dB	9.1dB	11.4dB	13.4dB	15.0dB	17.0dB	19.4dB	25.1dB

Figure 5: Average SNR_m improvement, from the initial SNR of clipped signals to the SNR of restored signals, as a function of the clipping level: for each dataset – *i.e.* each subfigure –, the performance is presented as a function of the clipping level, for OMP-G, OMP-G-min, OMP-G-minmax and OMP-C-minmax.

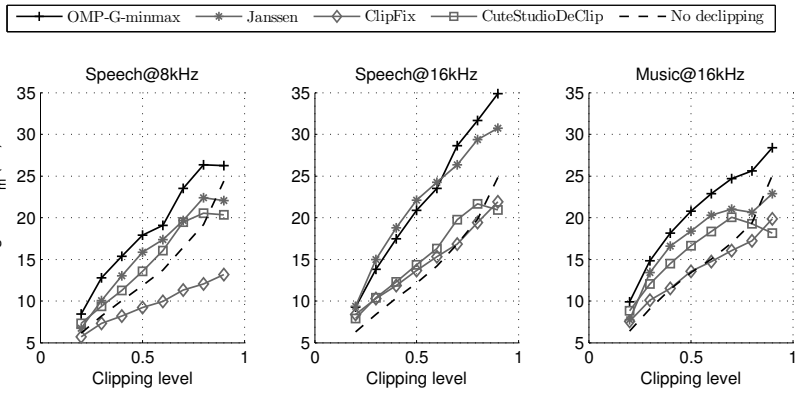


Figure 6: Average SNR_m as a function of the clipping level: for each dataset, the performance is presented as a function of the clipping level, for BOMP-minmax, for Janssen's approach [2] and for the spline interpolation ("Spline"). The initial SNR of the clipped signal is also plotted ("Clipped").

methods for audio declipping.

Based on the audio inpainting framework and on the baseline results presented in this paper, a number of future directions may be investigated. Technically, one may compare the OMP-based methods to l_1-minimization techniques, known to be another family of approaches to deal with sparse models. They are theoretically efficient but so far, we can only report preliminary performance that is lower than with greedy algorithms for audio inpainting. Another perspective is the use of new sparse models for audio signals. In particular, structured sparse models and learned dictionary are promising directions. From an application point of view, time-frequency audio inpainting is a new investigation field for sparse approaches. Using the formulation of audio inpainting (see Section 2.1) in the time-frequency domain, one must then introduce new dictionaries, targetting applications like source separation and bandwidth extension.

References

[1] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, "A Constrained Matching Pursuit Approach to Audio Declipping," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, May 2011.

[2] A. Janssen, R. Veldhuis, and L. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *IEEE Trans. Acoustics, Speech and Sig. Proc.*, vol. 34, no. 2, pp. 317 – 330, apr 1986.

[3] W. Etter, "Restoration of a discrete-time signal segment by interpolation based on the left-sided and right-sided autoregressive parameters," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1124 –1135, may 1996.

[4] S. J. Godsill and P. J. W. Rayner, *Digital Audio Restoration - A Statistical Model-based Approach*. Springer-Verlag, 1998.

[5] M. Lagrange and S. Marchand, "Long interpolation of audio signals using linear prediction in sinusoidal modeling," *Journal of the Audio Eng. Soc.*, vol. 53, pp. 891–905, 2005.

[6] A. Dahimene, M. Noureddine, and A. Azrar, "A simple algorithm for the restoration of clipped speech signal," *Informatica*, vol. 32, pp. 183–188, 2008.

[7] J. S. Abel and J. O. Smith, "Restoring a clipped signal," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Toronto, Canada, May 1991.

[8] R. C. Maher, "A method for extrapolation of missing digital audio data," in *95th AES Convention*, 1993.

[9] M. Cooke, P. Green, L. Josifovski, and A. Vizinho, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, vol. 34, no. 3, pp. 267 – 285, 2001.

[10] J. Gemmeke, H. Van Hamme, B. Cranen, and L. Boves, "Compressive sensing for missing data imputation in noise robust speech recognition," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 272 –287, 2010.

[11] J. Le Roux, H. Kameoka, N. Ono, A. de Cheveigné, and S. Sagayama, "Computational auditory induction as a missing-data model-fitting problem with Bregman divergence," *Speech Communication*, vol. In Press, 2010.

[12] M. Dietz, L. Liljeryd, K. Kjorling, and O. Kunz, "Spectral Band Replication, a novel approach in audio coding," in *Proc. of the 112th AES Convention*. Munich, Germany: Audio Engineering Society; 1999, May 2002.

[13] E. Larsen and R. Aarts, *Audio bandwidth extension: application of psychoacoustics, signal processing and loudspeaker design*. Wiley, 2004.

[14] P. Smaragdis, B. Raj, and M. Shashanka, "Missing data imputation for spectral audio signals," in *Proc. of MLSP*, Grenoble, France, Sep. 2009.

[15] M. Moussallam, P. Leveau, and S. M. Aziz Sbai, "Sound enhancement using sparse approximation with speclets," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Mar. 2010, pp. 221 –224.

[16] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *Network, IEEE*, vol. 12, no. 5, pp. 40 –48, sep. 1998.

[17] H. Ofir, D. Malah, and I. Cohen, "Audio packet loss concealment in a combined mdct-mdst domain," *Signal Processing Letters, IEEE*, vol. 14, no. 12, pp. 1032 –1035, dec. 2007.

[18] G. Cocchi and A. Uncini, "Subband neural networks prediction for on-line audio signal recovery," *IEEE Transactions on Neural Networks,*, vol. 13, pp. 867 – 876, 2002.

[19] S. J. Godsill, P. J. Wolfe, and W. N. W. Fong, "Statistical model-based approaches to audio restoration and analysis," *Journal of New Music Research*, vol. 30, no. 4, 2001.

[20] J. Barker, *Computational auditory scene analysis: principles, algorithms, and applications*. IEEE Press/Wiley-Interscience, 2006, ch. Robust Automatic Speech Recognition, pp. 297–350.

[21] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. of 27th Conf. on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 417–424.

[22] M. Elad, J. L. Starck, P. Querre, and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (mca)," *Applied and Computational Harmonic Analysis*, vol. 19, pp. 340–358, 2005.

[23] M. Elad, *Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing*. Springer New-York, 2010.

[24] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. On Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[25] M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. Davies, "Sparse representations in audio and music: from coding to source separation," *Proc. of the IEEE*, vol. 98, no. 6, 2010.

[26] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. On Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[27] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. on Image Processing*, vol. 17, no. 1, pp. 53 –69, 2008.

[28] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoustics, Speech and Sig. Proc.*, vol. 32, no. 2, pp. 236–243, Apr. 1984.

[29] M. Davies and L. Daudet, "Sparse audio representations using the MCLT," *Signal Processing*, vol. 86, no. 3, pp. 457 – 470, 2006.

[30] E. Ravelli, G. Richard, and L. Daudet, "Union of mdct bases for audio coding," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1361 –1372, nov. 2008.

[31] B. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Computing*, vol. 25, no. 2, pp. 227–234, 1995.

[32] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constr. Approx.*, vol. 13, no. 1, pp. 57–98, 1997.

[33] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proc. Asilomar Conf. Signals, Systems, and Computers*, Nov. 1993, pp. 40 –44 vol.1.

[34] R. Gribonval, "Fast matching pursuit with a multiscale dictionary of gaussian chirps," *IEEE Trans. on Signal Processing*, vol. 49, no. 5, pp. 994 –1001, May 2001.

[35] L. Peotta and P. Vandergheynst, "Matching pursuit with block incoherent dictionaries," *Signal Processing, IEEE Transactions on*, vol. 55, no. 9, pp. 4549 –4557, 2007.

[36] R. Gribonval and E. Bacry, "Harmonic decomposition of audio signals with matching pursuit," *IEEE Trans. Signal Processing*, vol. 51, no. 1, pp. 101 – 111, 2003.

[37] Y. Eldar and H. Bolcskei, "Block-sparsity: Coherence and efficient recovery," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, 2009, pp. 2885 –2888.

[38] H. Mansour, R. Saab, P. Nasiopoulos, and R. Ward, "Color image desaturation using sparse reconstruction," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Dallas, TX, USA, Mar. 2010.

[39] *Method for objective measurements of perceived audio quality*, ITU-R Std. BS.1387, Dec. 1998.

[40] R. Huber and B. Kollmeier, "Pemo-q - a new method for objective audio quality assessment using a model of auditory perception," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 1902 –1911, 2006.

[41] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, "Subjective and objective quality assessment of audio source separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. in press, 2011.

[42] E. Vincent, S. Araki, and P. Bofill, "The 2008 signal separation evaluation campaign: A community-based approach to large-scale evaluation." Paraty, Brazil: Springer, Mar. 2009.

[43] D. Donoho, Y. Tsaig, I. Drori, and J. Starck, "Sparse Solution of Underdetermined Linear Equations by Stagewise Orthogonal Matching Pursuit," *preprint*, 2006.

Contents

Centre de recherche INRIA Rennes – Bretagne Atlantique
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399