Audio inpainting based on joint-sparse modeling

Ichrak Toumi, Valentin Emiya

Audio inpainting based on joint-sparse modeling

Ichrak Toumi and Valentin Emiya *Member, IEEE,*

***Abstract*—We present a new framework for the restoration of missing samples in audio signals. It consists in locating audio frames that share similar sparse structures and in applying a joint-sparse algorithm to estimate the missing samples. Such similar frames are found in audio signals due to the signals' intrinsic structures: across channels, in the temporal neighboring of each frame and, since patterns are repeated non-locally. We propose a fast and robust strategy for locating the similar frames by introducing a spectral cosine similarity that is more suitable than the usual correlation similarity. We present and compare the inpainting versions of three known joint-sparse algorithms and show how they lead to a better reconstruction of the missing parts. Experimental results reveal that by selecting only a few similar frames, joint-sparse audio inpainting outperform the state-of-the-art OMP inpainting method by up to 5 dB, and that improvements cumulatively result from non-local and inter-channel joint decomposition.**

***Index Terms*—Audio inpainting, sparse approximation, joint sparsity, matching pursuit.**

I. Introduction

The audio inpainting framework aims at recovering the distored or missing parts in an audio signal based on the observed parts. In [1], the inpainting problem is formulated as an inverse problem in each audio frame. An inpainting algorithm based on a sparse decomposition was introduced, allowing the estimation of missing samples in audio frames from the approximated sparse support of the observed samples in a Gabor dictionary. The approach achieved competitive or better results compared to state-of-the-art methods like [2].

However, methods such as [1] process each audio frame independently while audio signals like speech, music and other sounds are highly structured across frames. As illustrated in Figure 1, this kind of structures originates from at least three phenomena. First, slowly-varying contents result in local similarities in the temporal neighborhood of each frame. Second, non-local similarities are observed since audio signals are generally composed of patterns that occur several times like phonemes and musical notes or chords. Third, simultaneous frames from multiple channels also offer several versions of some common contents with only small gain and delay differences. In the case of sparse models, similarity is defined in terms of a common sparse support, as illustrated by vertical arrays in Figure 1, with white elements representing zero entries. Our research hypothesis is that the use of these similar frames is advantageous to better estimate model parameters and solve inverse problems. The idea have been already used in image processing [3] and has been made possible using joint (a.k.a. simultaneous) sparse models [4], [5]: the authors assume that the similar image patches share the same supports and they apply an algorithm for joint-sparse decomposition.

I. Toumi and V. Emiya are with Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France (e-mail: firstname.lastname@lis-lab.fr).

This work was supported by ANR JCJC program MAD (ANR-14-CE27-0002).

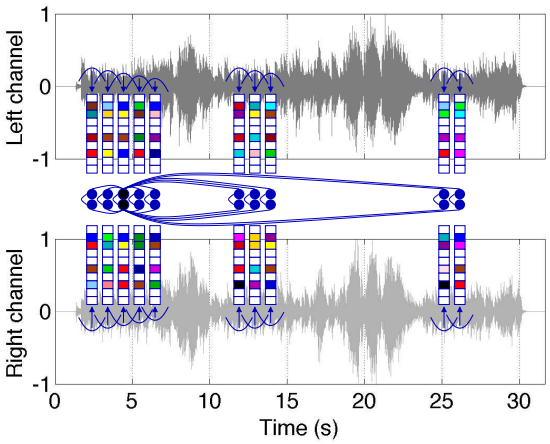

Fig. 1. Illustration of the joint-sparse structure of an audio signal: a target frame (black dot in the left part) has similar joint-sparse frames (in blue) identified simultaneously across channels (left/top and right/bottom), in its neighboorhood (selected frames in the left part) and non-locally (selected frames in the center and right parts).

Recently, this idea has motivated several works in audio processing [6]–[9]. It has proved for audio declipping [6] where dependencies between neighboring coefficients are exploited to improve the declipping performance. The results confirm the interest in using similarity information for inpainting clipped data. This approach being limited to the joint modeling of neighboring frames, finding and exploiting non-local similar frames is not addressed. Another strategy is proposed in [7] where repeated patterns are tracked in audio signals. This is performed in the particular case where the audio background is characterized by patterns repeated with a repeating period while the foreground has no repetition pattern. Non-local structures have been tracked using similarity graphs in [8], [9] for inpainting large gaps in audio signals. In [9], the inpainting of a large hole is achieved by comparing features extracted before and after the gap to other regions in order to find a region with similar contents to be copy-pasted inside the hole with smoothed transitions. In [8], more processing is proposed to adapt the contents of the pasted block with pitch, gain and time modifications, in the context of packet loss concealment.

In this paper, we propose to exploit similarity in audio signals in a sparse inpainting setting, in the continuity of [1], and extending preliminary works [10]. Given that the more missing samples, the less observations, the worse the sparse estimation, our research hypothesis is that finding similar regions for joint-sparse decomposition will provide more observation in order to improve the estimation quality and the resulting inpainting

performance. In such a context, the two main questions we address are:

1) In an audio signal with missing samples, how to locate regions that are similar in the sense that their latent sparse representations share a common support?
2) Given a set of regions similar to a target frame, which algorithms are appropriate for reconstructing the missing samples in this frame?

This paper is organized as follows. The general processing framework and notations are introduced in section II. In section III, we introduce the joint-sparsity approach and extend several existing algorithms to deal with missing data by using frame-dependent dictionaries. In section IV, we study how to select similar frames, based on a joint-sparse model structuring audio signals. Extensive experiments are reported in section V and conclusions are drawn in section VI.

II. Frame-based inpainting framework

We consider an audio signal $\underline{\mathbf{s}} \in \mathbb{R}^{\underline{L} \times C}$ of length $\underline{L}$ with C channels. A known binary mask $\underline{\mathbf{m}} \in \{0,1\}^{\underline{L} \times C}$ is used to locate missing and observed samples, coded by 0 and 1 entries respectively, so that the observation is $\underline{\mathbf{y}} = \underline{\mathbf{m}} \odot \underline{\mathbf{s}}$ where $\odot$ is the Hadamard product.

Signal $\underline{\mathbf{s}}$ is segmented into $N \triangleq \left\lfloor \frac{\underline{L}-L}{h} \right\rfloor$ overlapping frames of length L. For frame index $n \in [N]$ and channel c, $\mathbf{s}_{n,c} = [\underline{\mathbf{s}}(nh+l,c)]_{l \in [L]} \in \mathbb{R}^L$ denotes the n^{th} frame in channel c, where h is the hop size and $[N] \triangleq \{0, \ldots, N-1\}$. The binary mask $\underline{\mathbf{m}}$ and the observation vector $\underline{\mathbf{y}}$ are segmented similarly as a set of frame masks $\{\mathbf{m}_{n,c}\}$ and a set of observed frames $\{\mathbf{y}_{n,c}\}$. In order to simplify notations, we will also use index $j = (n,c)$ as a double index for a frame $\mathbf{s}_j$, the related mask $\mathbf{m}_j$ and observation $\mathbf{y}_j$.

Each frame is inpainted as a target frame, by means of the selection of other similar frames: the inpainting of a target frame given a set of similar frames is described in section III while the principle to select a set of similar frames is detailed in section IV. The restored signal can then be obtained from the reconstructed frames by a regular overlap-add procedure.

III. Joint-sparse audio inpainting

Starting from the formulation of the inpainting problem with a regular sparse model in section III-A, we introduce the formulation of the joint-sparse inpainting problem in section III-B. We propose several algorithms to obtain the joint-sparse decomposition from a set of frames with missing entries in section III-C.

A. Sparse problem formulation

In [1], each audio frame $\mathbf{s}$ is modeled using a sparse representation [11] as:

$$\mathbf{s} = \mathbf{D}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{D} \in \mathbb{R}^{L \times K}$ is the so-called dictionary, $\mathbf{x} \in \mathbb{R}^K$ is the sparse representation and $\mathbf{n} \in \mathbb{R}^L$ is the noise. Given a mask $\mathbf{m} \in \{0,1\}^L$ for frame $\mathbf{s}$ in the inpainting context, the observation is $\mathbf{y} = \mathbf{m} \odot \mathbf{s}$ and the inpainting optimization problem writes:

$$\arg\min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{m} \odot \mathbf{D}\mathbf{x}\|_2^2 < \epsilon \tag{2}$$

where $\epsilon > 0$ is a tolerance on the residual energy.

Since the ℓ_0 norm leads to an NP-hard problem, an approximated sparse solution may be obtained using a variant of the OMP algorithm [12] where all the dictionary columns are internally re-normalized to unit norm due to the partial observations. The unknown samples are then recovered from the given sparse solution $\widehat{\mathbf{x}}$ by reconstructing the frame as $\widehat{\mathbf{s}} \triangleq \mathbf{D}\widehat{\mathbf{x}}$.

B. Joint-sparse problem formulation

For a target frame $\mathbf{s}$, let us assume that we have an index set $\mathcal{S}$ so that frame $\mathbf{s}$ is included in the set of frames indexed by $\mathcal{S}$ and the sparse decompositions $\mathbf{x}_j$ of frames $\mathbf{s}_j, j \in \mathcal{S}$ have the same support. Many algorithms have been designed to obtain a joint-sparse decomposition $\mathbf{X}_\mathcal{S} = [\mathbf{x}_j]_{j \in \mathcal{S}}$ (e.g., see [13]–[16]). Joint-sparsity is typically promoted by minimizing the $\ell_{p,q}$ mixed (pseudo-)norm $\|\mathbf{X}_\mathcal{S}\|_{p,q}$ of matrix $\mathbf{X}_\mathcal{S}$ where the pair (p, q) usually takes the values $(0, \infty)$ to count the number of non-zero rows or $(1, 2)$ for a convex relaxation promoting joint sparsity.

In the inpainting case, the optimization problem can be formulated as

$$\arg\min_{\mathbf{X}_\mathcal{S}} \|\mathbf{X}_\mathcal{S}\|_{0,\infty} \ \text{s.t.} \ \|\mathbf{Y}_\mathcal{S} - \mathbf{M}_\mathcal{S} \odot \mathbf{D}\mathbf{X}_\mathcal{S}\|_F^2 < \epsilon_\mathcal{S} \tag{3}$$

where $\epsilon_\mathcal{S} > 0$, $\mathbf{Y}_\mathcal{S} \triangleq [\mathbf{y}_j]_{j \in \mathcal{S}}$ is the observation matrix in which each column is a selected observed frame $\mathbf{y}_j = \mathbf{m}_j \odot \mathbf{s}_j$, $\mathbf{M}_\mathcal{S} \triangleq [\mathbf{m}_j]_{j \in \mathcal{S}}$ is the binary mask matrix, and $\|\mathbf{X}_\mathcal{S}\|_{0,\infty} \triangleq \left| \bigcup_{j \in \mathcal{S}} \text{supp}(\mathbf{x}_j) \right|$ counts the number of non-zero rows in $\mathbf{X}_\mathcal{S}$.

C. Algorithms for joint-sparse inpainting

We propose three algorithms for joint-sparse decomposition by extending their original formulation to the inpainting framework. The extension consists in building frame-dependent dictionaries to handle missing data as described in section III-C1. The three algorithms are detailed in sections III-C2, III-C3 and III-C4. The reconstruction of the frames is eventually described in section III-C5

1) Frame-dependent dictionaries: Since each selected frame has its own mask, one must use frame-dependent dictionaries instead of a unique dictionary for all frames. For $j \in \mathcal{S}$, the normalized dictionary for frame j is defined as

$$\mathbf{D}_j \triangleq \text{diag}(\mathbf{m}_j) \times \mathbf{D} \times \mathbf{W}_j \tag{4}$$

where $\mathbf{W}_j \triangleq \text{diag}\left(\left[\|\mathbf{m}_j \odot \mathbf{d}_k\|_2^{-1} \right]_{k \in [K]} \right)$ is the normalization matrix. In eq. (4), the left product with $\text{diag}(\mathbf{m}_j)$ results in zeroing the rows of $\mathbf{D}$ related to missing coefficients and the right product by $\mathbf{W}_j$ leads to atoms with unit l_2-norms. Problem (3) can then be rewritten as

$$\arg\min_{\mathbf{X}'_\mathcal{S}} \|\mathbf{X}'_\mathcal{S}\|_{0,\infty} \ \text{s.t.} \ \sum_{j \in \mathcal{S}} \left\|\mathbf{y}_j - \mathbf{D}_j \mathbf{x}'_j\right\|_2^2 < \epsilon_\mathcal{S} \tag{5}$$

whose solution $\mathbf{X}'_{\mathcal{S}} = \left[\mathbf{x}'_j\right]_{j\in\mathcal{S}}$ is related to the solution $\mathbf{X}_{\mathcal{S}} = [\mathbf{x}_j]_{j\in\mathcal{S}}$ of (3) by the rescaling

$$\mathbf{x}_j = \mathbf{W}_j \mathbf{x}'_j. \tag{6}$$

2) Inpainting S-OMP: The Simultaneous Orthogonal Matching Pursuit (S-OMP) generalizes the OMP algorithm to the joint-sparsity case [14]. The inpainting version of S-OMP is described in Algorithm 1. It mainly consists in building the joint-sparse support Γ^i in a greedy way, starting from the empty set (line 5) and adding one atom at each iteration (line 10). The atom is selected jointly across frames by adding up the absolute correlations between each frame residual and the frame-related atoms (lines 8 and 9). The residual is then updated independently for each frame (line 11).

Algorithm 1 S-OMP inpainting algorithm.

Inputs: $\mathbf{Y}_{\mathcal{S}}$, $\mathbf{M}_{\mathcal{S}}$, $\mathbf{D} = [\mathbf{d}_k]_{k\in[K]}$, $i_{\max}$, $\epsilon_{\mathcal{S}}$

1: **for** $j \in \mathcal{S}$ **do**
2: Build frame-dependent normalized dictionary $\mathbf{D}_j$ and normalization matrix $\mathbf{W}_j$ from $\mathbf{D}$ and $\mathbf{m}_j$ using eq. (4).
3: **end for**
4: Iteration counter $i = 0$
5: Support $\Gamma^0 = \emptyset$
6: Residual $\mathbf{r}_j^0 = \mathbf{y}_j, \forall j \in \mathcal{S}$
7: **while** $i < i_{\max}$ and $\sqrt{\sum_{j\in\mathcal{S}} \|\mathbf{r}^i\|_2^2} \geq \epsilon_{\mathcal{S}}$ **do**
8: $\mathbf{c}_j = \mathbf{D}_j^H \mathbf{r}_j^i, \forall j \in \mathcal{S}$
9: Select atom $\widehat{k} = \underset{k\in[K]}{\arg\max} \sum_{j\in\mathcal{S}} |\mathbf{c}_j(k)|$
10: Update support $\Gamma^{i+1} = \Gamma^i \cup \left\{\widehat{k}\right\}$
11: Update current solution and residuals $\mathbf{r}_j^{i+1} = \left(\mathbf{I} - \mathbf{D}_{j,\Gamma^{i+1}} \mathbf{D}^+_{j,\Gamma^{i+1}}\right) \mathbf{y}_j, \forall j \in \mathcal{S}$
12: $i = i + 1$
13: **end while**
14: **for** $j \in \mathcal{S}$ **do**
15: Create output vectors: $\mathbf{x}_j \triangleq \mathbf{0}_K$
16: Compute sparse representation: $\mathbf{x}_j\left(\Gamma^i\right) \triangleq \mathbf{D}^+_{j,\Gamma^i}\mathbf{y}_j$
17: Correct dictionary normalization: $\mathbf{x}_j = \mathbf{W}_j\mathbf{x}_j$
18: **end for**
19: **return** $\mathbf{X}_{\mathcal{S}} \triangleq [\mathbf{x}_j]_{j\in\mathcal{S}}$

3) Inpainting S-IHT: The simultaneous Iterative Hard Thresholding algorithm (S-IHT) is an extension of the Iterative Hard Thresholding (IHT) algorithm to the case of joint sparse signals [13], [17]. The inpainting version given in Algorithm 2 relies on iteratively refining a sparse support Γ^i and joint T-sparse vectors $\mathbf{x}_j^i$. The sparse support is recomputed at each iteration, jointly across frames, by performing a gradient step (line 9), by adding up the absolute values of the result for all frames (10) and by retaining the location of T highest entries (11). The update of the sparse representations is obtained independently for each frame by considering the gradient step in each frame and by keeping only the values indexed by the support Γ^i, using the hard-thresholding operator $H_{\Gamma^{i+1}}$ (line 13). A faster version of the algorithm has been proposed in [17] where the step size is an adaptative scaling factor μ_j^i for each selected frame k and at each iteration i (line 15).

Algorithm 2 S-IHT inpainting algorithm.

Inputs: $\mathbf{Y}_{\mathcal{S}}$, $\mathbf{M}_{\mathcal{S}}$, $\mathbf{D} = [\mathbf{d}_k]_{k\in[K]}$, T, $i_{\max}$, $\epsilon_{\mathcal{S}}$

1: **for** $j \in \mathcal{S}$ **do**
2: Build frame-dependent normalized dictionary $\mathbf{D}_j$ and normalization matrix $\mathbf{W}_j$ from $\mathbf{D}$ and $\mathbf{m}_j$ using eq. (4).
3: **end for**
4: Iteration counter $i = 0$
5: Support $\Gamma^0 = \emptyset$
6: Sparse coefficients $\mathbf{x}_j^0 = 0$
7: adaptative scaling factor $\mu_j^0 = \frac{1}{L}$
8: **while** $i < i_{\max}$ and $\sum_{j\in\mathcal{S}} \left\|\mathbf{y}_j - \mathbf{D}_{j\Gamma^i}\mathbf{x}_j^i\right\|_2 > \epsilon_{\mathcal{S}}$ **do**
9: $\mathbf{z}_j^{i+1} = \left|\mathbf{x}_j^i + \mu_j^i \mathbf{D}_j^T(\mathbf{y}_j - \mathbf{D}_k\mathbf{x}_k^i)\right|, \forall j \in \mathcal{S}$
10: $\mathbf{w}^{i+1} = \frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}} \mathbf{z}_j^{i+1}$
11: $\Gamma^{i+1} = \left\{\text{indices of T largest coefficients of } \mathbf{w}^{i+1}\right\}$
12: **for** $j \in \mathcal{S}$ **do**
13: $\mathbf{x}_j^{i+1} = H_{\Gamma^{i+1}}\left(\mathbf{x}_j^i + \mu_j^i \mathbf{D}_j^T(\mathbf{y}_j - \mathbf{D}_j\mathbf{x}_j^i)\right)$
14: $\mathbf{g}_j = \mathbf{D}^T_{j,\Gamma^{i+1}}(\mathbf{y}_j - \mathbf{D}_{j,\Gamma^{i+1}}\mathbf{x}_j^i)$
15: $\mu_j^{i+1} = \frac{\mathbf{g}^T_{j,\Gamma^{i+1}}\mathbf{g}_{j,\Gamma^{i+1}}}{\mathbf{g}^T_{j,\Gamma^{i+1}}\mathbf{D}^T_{j,\Gamma^{i+1}}\mathbf{D}_{j,\Gamma^{i+1}}\mathbf{g}_{j,\Gamma^{i+1}}}$
16: **end for**
17: $i = i + 1$
18: **end while**
19: **for** $j \in \mathcal{S}$ **do**
20: Create output vectors: $\mathbf{x}_j \triangleq \mathbf{0}_K$
21: Compute sparse representation: $\mathbf{x}_j\left(\Gamma^i\right) \triangleq \mathbf{D}^+_{j,\Gamma^i}\mathbf{y}_j$
22: Correct dictionary normalization: $\mathbf{x}_j = \mathbf{W}_j\mathbf{x}_j$
23: **end for**
24: **return** $\mathbf{X}_{\mathcal{S}} \triangleq [\mathbf{x}_j]_{j\in\mathcal{S}}$

4) Inpainting S-CoSaMP: The third algorithm is the extension of greedy algorithm CoSaMP [16] to the joint-sparse case [13], [18]. The inpainting version is given in Algorithm 3. A sparse support Γ^i is iteratively refined, by extending it (line 11) with indexes of νT atoms that best correlate with the residuals, in a joint way across frames (lines 9 and 10). An orthogonal projection of the observed frames onto the extended set of atoms is then performed (lines 13 and 14) and the support update is then obtained by selecting the T largest coefficients (line 16). The residual update is eventually performed independently for each frame (line 17), similarly as in the S-OMP algorithm.

The extension of the support at each iteration is controled by the positive hyperparameter ν that must be adjusted by the user. In our case, a small value of ν is recommended to ensure a better stability of the pseudo-inverse at line 13.

5) Frame reconstruction: Each algorithm returns an approximate solution $\widehat{\mathbf{X}}_{\mathcal{S}}$ to problem (3). Since $\mathcal{S}$ contains the indexes of a target frame and of similar frames, one may only use the sparse representation $\widehat{\mathbf{x}}$ to reconstruct the target frame as $\widehat{\mathbf{s}} \triangleq \mathbf{D}\widehat{\mathbf{x}}$, discarding the other columns of $\widehat{\mathbf{X}}_{\mathcal{S}}$.

IV. Selecting frames for joint-sparse decompositions

Our approach for joint-sparse decomposition and inpainting requires the prior selection of similar frames, which are identified by the index set $\mathcal{S}$. Here, similarity should be understood

Algorithm 3 S-CoSaMP inpainting algorithm.

Inputs: $\mathbf{Y}_{\mathcal{S}}$, $\mathbf{M}_{\mathcal{S}}$, $\mathbf{D} = [\mathbf{d}_k]_{k\in[K]}$, T, ν, $i_{\max}$, $\epsilon_{\mathcal{S}}$

for $j \in \mathcal{S}$ **do**
  Build frame-dependent normalized dictionary $\mathbf{D}_j$ and normalization matrix $\mathbf{W}_j$ from $\mathbf{D}$ and $\mathbf{m}_j$ using eq. (4).
end for
Iteration counter $i = 0$
Support $\Gamma^0 = \emptyset$
Sparse coefficients $\mathbf{x}_j^0 = 0$
Residual $\mathbf{r}_j^0 = \mathbf{y}_j, \forall j \in \mathcal{S}$
while $i < i_{\max}$ and $\sum_{j\in\mathcal{S}} \left\|\mathbf{r}_j^i\right\|_2 > \epsilon_{\mathcal{S}}$ **do**
  $\mathbf{c}_j = \mathbf{D}_j^H \mathbf{r}_j^i, \forall j \in \mathcal{S}$
  Select support $\Omega = \underset{|\Omega|\leq \nu T}{\arg\max} \sum_{k\in\Omega}\sum_{j\in\mathcal{S}} \left|\mathbf{c}_j(k)\right|^2$
  $Q^i = \Gamma^i \cup \Omega$
  for $j \in \mathcal{S}$ **do**
    $\mathbf{b}_j\left(Q^i\right) = \mathbf{D}^+_{j,Q^i}\mathbf{y}_j$
    $\mathbf{b}_j\left(\overline{Q^i}\right) = 0)$
  end for
  Select support $\Gamma^{i+1} = \underset{|\Gamma|=T}{\arg\max} \sum_{k\in\Gamma}\sum_{j\in\mathcal{S}} \left|\mathbf{b}_j(k)\right|^2$
  $\mathbf{r}_j^{i+1} = \left(\mathbf{I} - \mathbf{D}_{j,\Gamma^{i+1}}\mathbf{D}^+_{j,\Gamma^{i+1}}\right)\mathbf{y}_j, \forall j \in \mathcal{S}$
  $i = i + 1$
end while
for $j \in \mathcal{S}$ **do**
  Create output vectors: $\mathbf{x}_j \triangleq \mathbf{0}_K$
  Compute sparse representation: $\mathbf{x}_j\left(\Gamma^i\right) \triangleq \mathbf{D}^+_{j,\Gamma^i}\mathbf{y}_j$
  Correct dictionary normalization: $\mathbf{x}_j = \mathbf{W}_j\mathbf{x}_j$
end for
return $\mathbf{X}_{\mathcal{S}} \triangleq [\mathbf{x}_j]_{j\in\mathcal{S}}$

in terms of a joint-sparse representation. In this section, we elaborate on how to build such a set. The main idea is that joint-sparsity is due to at least three phenomena:

1) in a multichannel recording (e.g., stereo), the contents in all channels is the same up to some variations that may not affect joint-sparsity;
2) the contents in successive frames is varying slowly compared to the frame length —which is generally chosen to obtain quasi-stationarity within frames;
3) multiple occurrences of the same audio item —such as phoneme or musical notes— are observed with significant delays between them.

As a result, we propose a triple joint-sparse model in which all those similarities are tracked. Interchannel joint sparsity is introduced in section IV-A in order to select similar frames from all channels. Joint sparsity in adjacent and non-local frames is described in section IV-B, resulting in a selection of frames at various frame indices.

A. Interchannel frame selection

Joint-sparsity across channels can be obtained if the sparse support is invariant to the actual interchannel differences. A global gain difference does not affect the sparse support in an undesirable way, since only the non-zero coefficients are modified, and possibly vanish. However in a more general, convolutive model of audio signal, the same sources are recorded in each channel after being filtered in a channel-dependent way, e.g., by Green functions or head-related transfer functions. As a result, the differences between channels are frequency-dependent gain and phase variations, which may affect the sparse support. Hopefully, choosing a Fourier dictionary is suitable since it is not sensitive to those variations, besides being adapted to the sparse decomposition of audio signals.

Hence, the multichannel setting is a beneficial way to gather joint-sparse frames, which can be done by systematically selecting frames from all channels and decomposing them jointly in a Fourier dictionary.

B. Neighboring and non-local frame selection

Let us fix the index n of a target frame. The selection of frames that are similar to this target frame in its temporal neighboring and in possibly-far regions of the signal requires a selection criterion. Given a similarity measure γ and a set of candidate frames indexed by $n' \in [N]$, one may build the index set of similar frames by selecting the most similar frames from all channels:

$$\mathcal{S} \triangleq \{(n,c)\,; n \in [N]\,, c \in [C]\,, \gamma(n,n') \geq \gamma_{\mathcal{S}}\} \qquad (7)$$

where parameter $\gamma_{\mathcal{S}}$ is adjusted to control the number of selected frames ζ which is equal to $|\mathcal{S}|$.

One important issue is the choice of a similarity measure that is suitable to the joint-sparse decomposition. The goal is to select a set of frames with sparse approximations sharing the same support as that of the target frame without having to compute their sparse decomposition. This requires a similarity measure that has the ability to act as a proxy to the comparison between the sparse supports. We propose and compare three different similarity criteria in order to find the most appropriate measure for a joint-sparse approximation in a redundant Fourier dictionary.

1) Proposed similarity measures: Similarity measures for the selection of non-local regions have been already used in image processing [3] and are generally based on the correlation between images patches, which would correspond to correlating audio frames in our setting. Starting from this criterion, we propose three different measures to assess the similarity between the target frame and a candidate frame, and study how they are suitable for a joint-sparse approximation in a redundant Fourier dictionary:

- the **correlation** may be viewed as the cosine of the angle between the K-point DFT $\widehat{\mathbf{y}}_{n,c}$ and $\widehat{\mathbf{y}}_{n',c}$ of both frames:

$$\begin{aligned}\gamma_{corr}(n,n') &\triangleq \sum_{c\in[C]} \left|\left\langle \frac{\mathbf{y}_{n,c}}{\|\mathbf{y}_{n,c}\|_2}, \frac{\mathbf{y}_{n',c}}{\|\mathbf{y}_{n',c}\|_2} \right\rangle\right| \\ &= \sum_{c\in[C]} \left|\left\langle \frac{\widehat{\mathbf{y}}_{n,c}}{\|\widehat{\mathbf{y}}_{n,c}\|_2}, \frac{\widehat{\mathbf{y}}_{n',c}}{\|\widehat{\mathbf{y}}_{n',c}\|_2} \right\rangle\right| \\ &= \sum_{c\in[C]} \left|\cos\left(\angle\left(\widehat{\mathbf{y}}_{n,c}, \widehat{\mathbf{y}}_{n',c}\right)\right)\right| \qquad (8)\end{aligned}$$

- the **spectral cosine similarity** discards phase effects and is computed as the cosine similarity between the normalized modulus of the DFT vectors:

$$\gamma_m(n,n') \triangleq \sum_{c\in[C]} \left|\cos\left(\angle\left(\left|\widehat{\mathbf{y}}_{n,c}\right|,\left|\widehat{\mathbf{y}}_{n',c}\right|\right)\right)\right| \quad (9)$$

- the **Itakura-Saito (IS) similarity** is defined as $\gamma_{IS}(n,n') \triangleq 1 - d_{IS}(n,n')$ from the IS divergence widely used for audio processing:

$$d_{IS}(n,n') \triangleq \frac{1}{CK}\sum_{c\in[C]}\sum_{k}\left[\left|\frac{\widehat{\mathbf{y}}_{n,c}(k)}{\widehat{\mathbf{y}}_{n',c}(k)}\right| - \log\left(\left|\frac{\widehat{\mathbf{y}}_{n,c}(k)}{\widehat{\mathbf{y}}_{n',c}(k)}\right|\right) - 1\right] \quad (10)$$

2) Selecting an appropriate similarity measure: We experimentally compare the ability of the proposed criteria to select the same frames with a reference method which computes the sparse decompositions of the frame supports to find the most similar ones (see Figure 2). We start by segmenting our signal into a set of overlapped frames $\{\mathbf{s}_j\}$ with a hop size h. Missing data are artificially created at random locations to obtain observed frames $\{\mathbf{y}_j\}$. Then, on one hand, a groundtruth is designed as follows: we extract a sparse decomposition of the frames using OMP algorithm and compute the Hamming distance between their supports $\{\mathbf{x}_j\}$, resulting in a similarity matrix $\mathbf{\Gamma}_{ham}$. On the other hand, we compute the similarity matrices $\mathbf{\Gamma}_{corr}$,$\mathbf{\Gamma}_m$,$\mathbf{\Gamma}_{IS}$ based on the different similarity measures presented above. For each target frame $\mathbf{y}$, the ζ most similar frames among all the candidate frames $\{\mathbf{y}_j\}$ are given by the ζ largest values for each measure. Then the intersection between the sets of indices ($\mathcal{S}_{corr}$, $\mathcal{S}_m$, $\mathcal{S}_{IS}$) and groundtruth $\mathcal{S}_{ham}$ is computed in order to select the best similarity measure.

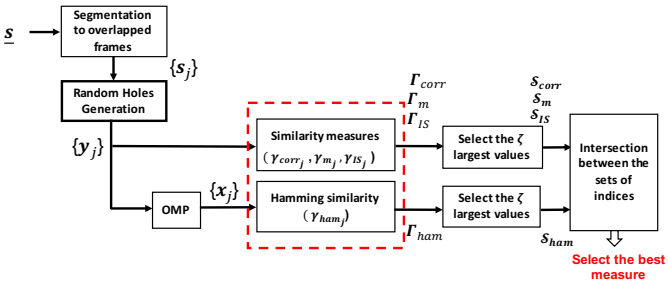

Fig. 2. Diagram of the method for the selection of the best similarity measure.

The process is repeated for different ratios of missing data, using the same experimental conditions as in Section V and for $\zeta = 4$. Results are averaged for a speech and a piano examples and are shown in Table I.

The results show that the correlation measure is not appropriate and that the spectral cosine similarity seems to be more efficient than the IS similarity for all the considered ratios of missing data. To further understand these quantitative results, similarity maps are generated for the reference method and for each proposed measure showing the selected frames and how they are located. In Figure 3, we give an example for a small region in a speech audio signal sampled at 8KHz.

Missing data (%)	0	20	40	60	80
γ_{corr}	29.2	28	27.8	26.9	26.3
γ_m	**39.9**	**35.9**	**33.8**	**32**	**29.6**
γ_{IS}	35.1	32.9	31.6	30.3	28.5

TABLE I
MEAN INTERSECTION BETWEEN THE SETS OF FRAMES SELECTED BY THE REFERENCE METHOD AND THE PROPOSED SIMILARITY MEASURES, AS A FUNCTION OF THE RATIO OF MISSING DATA. THE MEAN INTERSECTIONS ARE GIVEN AS A PERCENTAGE OF THE TOTAL NUMBER OF SELECTED FRAMES.

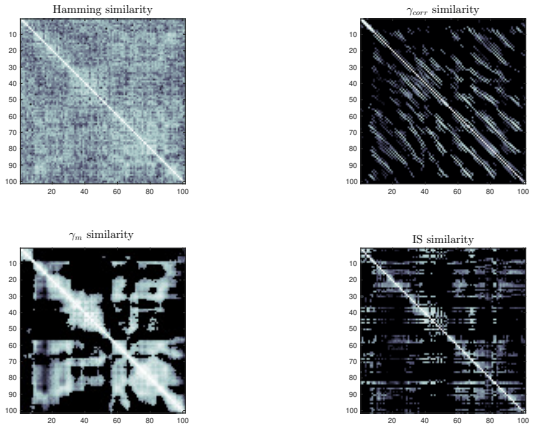

Fig. 3. Similarity matrices for the reference frame selection method (Hamming similarity, top left) and the proposed measures.

We can see that the structures in the Hamming map are approximately reproduced in the similarity maps of the other measures except the correlation in which aligned structures parallel to the diagonal appear. This is explained by sensitivity of the correlation measure to the interferences due small phase differences which may cause low correlation values even when frames have similar sparse supports which is not the case for the other studied measures. Concretely, correlation in our case could be relevant when only using a unit hop size to select similar frames that are perfectly aligned, which would be computationally demanding and not necessary for the joint sparsity algorithm using a Fourier dictionary. Hence, the spectral cosine similarity happens to be the most appropriate measure and we select it for the experiments conducted in section V.

V. EXPERIMENTAL RESULTS

We present the results of our approach based on the frame selection strategy combined with the joint-sparse optimization. In section V-A, we introduce the experimental setting, including the audio material, problem generation and frame-based model parameters. In the section V-B, the tuning of two sensitive hyperparameters is described. Then, in section V-C, we show how the method can solve an audio inpainting problem for different rates of missing data[1]. We compare the baseline OMP inpainting algorithm [1] and the proposed

[1]For reproducibility, the code and data are available at https://mad.lis-lab.fr/.

inpainting versions of the joint-sparse algorithms S-OMP, S-IHT and S-COSAMP, considering the single-channel case ($C = 1$) and the multichannel case ($C = 2$).

A. Experimental setting

1) Audio material: To validate the non local frame selection process and tune hyperparameters, we use two audio signals of duration 4 s sampled at 8 kHz: a melody piano and a male speech composed of one sentence. The inpainting methods are evaluated over a larger set of other sounds from the SQAM dataset[2]. We have selected 26 sound excerpts that are classified into three categories, Speech, Music solo and Music group, whose main features are summarized in Table II.

Category	Number of sounds	Content	Sampling frequency	Mean duration
Speech	6	Male & female sentences	8 kHz	7 s
Music solo	9	Single instruments & vocals	16 kHz	10 s
Music group	11	Pop music & Vocal with Orchestra	16 kHz	11 s

TABLE II
CONTENTS OF THE DATASET USED FOR THE INPAINTING EVALUATION.

2) Problem generation and performance measures: the missing samples are generated randomly and uniformly according over the whole signal. The protocol consists in fixing the number of missing samples L_{miss}. Then for each $(a, b) \in \mathbb{N}$ such that $a \times b = L_{miss}$, a holes with length b are generated randomly for each frame in the audio signal. The performance of the inpainting methods is assessed for a missing data rate ranging from 10% to 90%.

The audio inpainting performance is evaluated in terms of signal-to-noise-ratio averaged over the reconstructed frames as defined in [1], either on all the samples ($\mathrm{SNR}_{\mathrm{full}}$) or on the recovered samples only (SNR_m). The proposed algorithms are compared to the inpainting OMP algorithm [1], considered as a baseline since it relies on a similar sparse model without joint-sparsity aspects.

3) Frame-based model: for all the audio signals, we set the frame length to $L = 256$ samples and the hop size to $h = \frac{L}{4}$. As a result, speech signals sampled at 8 kHz are segmented with a hop size of 8 ms to get around 900 frames with length 32 ms, while for music signals, the number of frames is about 2600 with a length of 16 ms and a hop size of 4 ms. The inpainting algorithms are compared using a redundant complex Fourier dictionary with a size 256×512.

B. Tuning hyperparameters

This section is dedicated to the tuning of the sparsity parameter T and the number of selected frames ζ. Those important parameters for the proposed approach are tuned using the piano and speech sounds previously mentioned. Beyond finding good values for those hyperparameters, the proposed tuning experiments are also the opportunity to illustrate the behavior of the algorithms and to comment on them.

1) Sparsity T: one important hyperparameter is the number of selected atoms T for the sparse modeling. We question the relevance to fix it to a constant value, regardless of the missing data ratio, or to make it depend on this feature. Indeed, when the number of observations is reduced due to missing data, one cannot hope to retrieve the whole sparse support and must resort to more regularization with low T values. We study the optimal values of T for each algorithm and for various missing data ratios p. In order to sample the (p, T) space adequately, the sparse optimization is carried out using values of T from a p-independent set $\left\{2^\ell; \ell \in \{3, \ldots, 7\}\right\}$ and from a p-dependent set $\left\{\left[(1 - 0.9\,p) \times 2^\ell\right]; \ell \in \{5, 5.5, \ldots, 8\}\right\}$, where $[.]$ is the rounding operator and p is the ratio of missing data. The optimal value T_{opt} is obtained from the best SNR_m value for each algorithm and every ratio p, with ζ is set to 4.

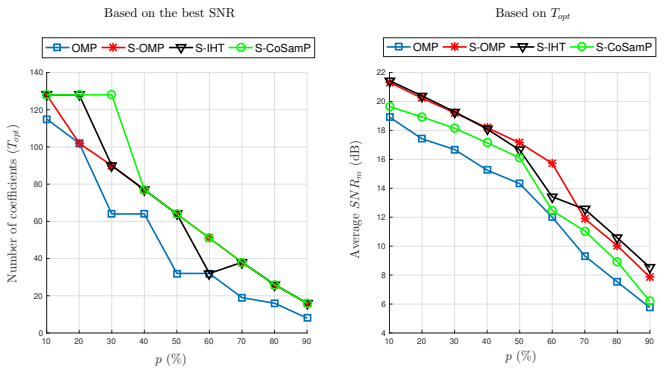

Fig. 4. Selection of T_{opt} for the sparse optimization algorithms according to the best SNR computed over the missing samples.

From left part of figure 4, we can see that T_{opt} decreases approximately linearly with respect to p. It also shows that the joint sparse algorithms give similar T_{opt} values for all ratios p. The right part of figure 4 give some preliminary SNR_m performance values for each algorithm, showing how the performance decreases as a function of the ratio of missing data. It suggests that S-OMP and S-IHT inpainting algorithms may perform better then OMP and S-CoSamP inpainting algorithms, which needs to be confirmed on a separate and larger set of sounds, as proposed in section V-C. The retained T values are summarized in Table III.

Missing data (%)	20	40	60	80
OMP	102	64	32	16
S-OMP	102	77	51	26
S-IHT	128	77	32	26
S-CoSamP	128	77	51	26

TABLE III
T VALUES RETAINED FOR THE DIFFERENT INPAINTING ALGORITHMS.

2) Number of similar frames ζ: another crucial parameter for the joint sparsity optimization is the size of $\mathcal{S}$ which corresponds to the number of similar frames ζ. In figure 5, the average SNR_m, computed for the speech and piano sounds using S-OMP algorithm, is plotted as a function of the number

[2]See https://tech.ebu.ch/publications/sqamcd.

of selected frames $|\mathcal{S}|$ for a particular case where $p = 50\%$ and holes duration is equal to 0.25 ms.

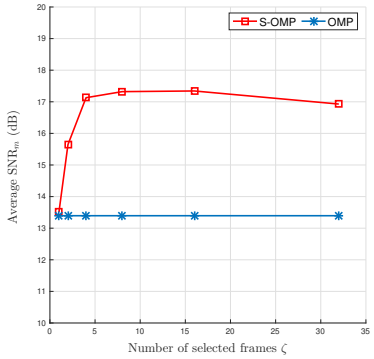

Fig. 5. Average SNR_m as a function of the number of selected frames for the joint sparse algorithms (ζ).

The study shows that good SNR values are obtained for a size $|\mathcal{S}|$ equal to or greater than 4. When $\zeta = 1$, the S-OMP inpainting performance is equivalent to that of OMP while $\forall\ \zeta > 1$ better performance is obtained, illustrating the improvement provided by multiple frame observations. For $\zeta \gg 4$, the SNR improvement may decrease since the additional selected frames may not be as similar as the first 4 selected frames. In addition, selecting a large set of frames may result in a large computational cost. Hence, for the rest of the experiments, we set $|\mathcal{S}|$ to 4.

This experiment gives also the opportunity to illustrate the origin of the selected frames. In particular, one may wonder whether the selected frames are neighboring frames located next to the target frame or whether they are very far away from the time of the target frame. Figure 6 illustrates the origin of the selected frames by counting the ratio of neighboring frames among the selected frames. On the absissa, a varying frame index radius is used as a threshold to separate the selected into neighboring frames and non-local frames.

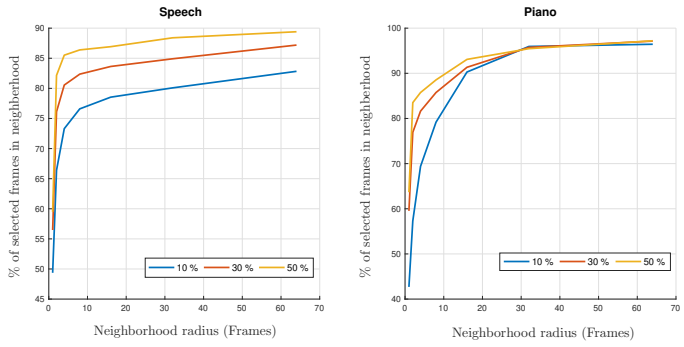

Fig. 6. Percentage of selected frames in neighborhood as a function of neighborhood radius; each curve is related to a ratio of missing data.

According to Figure 6, between 60 and 90% of the selected frames are located in the neighborhood of the target frames, depending on the ratio of missing data and the sound contents. It clearly appears that the more missing data, the more local selection. However, the number of non-local selected frames is significant, which demonstrates the utility to search for non-local similarities, as proposed in this paper.

Table IV finally summarizes the retained parameters.

Parameter	Value
Frame length (L)	32 ms @ 8 kHz and 16 ms @ 16 KHz
Hop size (h)	$\frac{L}{4}$
Dictionary size	Ω = 2 $\times$ L = 512
T	see Table (III)
ζ	4
Block duration	0.25 ms @ 8 kHz and 0.125 ms @ 16 kHz
ν	0.25
Iterations number (i_{max})	500

TABLE IV
SUMMARY OF PARAMETER SETTINGS.

C. Inpainting results

Inpainting performance is reported for mono and stereo sounds separately in order to illustrate how the performance is impacted by the origin of the similar frames (neighboring and non-local vs. interchannel).

1) Mono sounds ($C = 1$): the evaluation of the inpainting results for single-channel sounds using the proposed algorithms is given in figures 7, 8 and 9. From all the figures, one can see that the inpainting algorithms based on the joint sparsity outperforms globally the inpainting OMP, specially in the reconstructed regions as illustrated by the SNR_m values. For the speech sounds in figure 7, the three presented inpainting versions of S-OMP, S-IHT and S-CoSamP have a better average SNR_m compared to the inpainting OMP, whatever the rate of missing data. However, when looking at the average $\mathrm{SNR}_{\mathrm{full}}$ for extremely low values of p ($< 20\%$), they are less efficient: for instance, the S-CoSamP algorithm does not outperfom OMP until ($p = 40\%$). This can be explained as follows: first, the number of observations is high enough so that OMP gives a good sparse decomposition. Second, generally, the joint sparse algorithms constrain the support of the selected frames to be equal, which may degrade the data fitting term (i.e., the reconstruction of the observations), especially with real data where supports are not exactly the same. In addition, the S-CoSamP algorithm is very sensitive to the pseudo-inverse. The stability of the latter depends on setting parameters like ν.

For music sounds in figures 8 and 9, we note the same behavior except that the improvement provided by the joint sparse algorithms is more important. This is reflected in a larger gap between the SNR values for S-OMP and S-IHT compared to OMP when p is comprised between $10\% < p < 70\%$. The improvement in SNR values with the joint sparse methods reaches up to 5 dB for the average $\mathrm{SNR}_{\mathrm{full}}$ and SNR_m values. Furthermore, we notice that the inpainting version of S-OMP is giving better performance than S-IHT, with music sounds, contrary to speech sounds for which the inpainting S-IHT outperforms the other algorithms. The difference in SNR between the two algorithms goes from 0 to 2 dB.

2) Stereo sounds ($C = 2$): in the following, we show how the performance may be additionally improved by taking into account the similarity between channels. The evaluation of the multichannel case is based on two hypotheses:

1) masks in each channel may be different;
2) similarity measures are computed channel-wise, as given by the sum over channels in eq. (8), (9) and (10).

In order to compare the mono case to the stereo case, the average SNR_{full} and SNR_m are computed using similar frames selected in two different ways:

- Mono case (solid lines): each channel is processed separately to select similar frames and apply a joint-sparse inpainting algorithm, and performance is averaged over channels.
- Stereo case (dashed lines): similar frames from both channel are selected jointly.

The results are reported in figures 10, 11 and 12, where the SNR values are plotted as a function of the ratio of missing samples. For each algorithm, the average SNR between the two channels is represented.

Globally, the results show that an additional improvement is obtained by considering frames from both channels. For speech signals (see figure 10), the SNR_m improvement brought by the use of a selection over the two channels is significant, reaching about 2 dB. Unlike for speech, the SNR improvement is not significant for music solo sounds (figure 11), and for music group sounds (figure 12), only the S-OMP inpainting algorithm shows some noticeable improvement of almost 1 dB.

By comparing the performance obtained in the mono and stereo cases, the results suggest that depending on the sound contents, the improvement obtained by the proposed joint-sparse approach may originate differently: for speech sounds, the global SNR_m improvement is about 2.5 dB and comes both from the frame selection in time and between channels, while for music sounds, the global improvement is up to 5 dB and is mainly due to the selection of neighboring and non-local frames, rather than from the interchannel observations. This may be due to the fact that music sounds can be composed of very similar occurrences of some patterns while speech contents may vary a lot in excerpts that last a few seconds.

Finally, one may retain that the S-OMP inpainting algorithm seems to give the best performance compared to the other joint-sparse inpainting algorithms S-IHT and S-CoSamP.

VI. Conclusion

In this paper, a new framework for audio inpainting using joint-sparsity is presented based on two steps: first, in the audio signal, similar sparse structures are selected using a similarity measure called *spectral cosine similarity* which proved to be suitable for the joint-sparse decomposition using a redundant Fourier dictionary. Second, a joint-sparse algorithm, using frame-dependent dictionaries, is applied to recover the missing samples. All the inpainting versions of the studied joint-sparse algorithms S-OMP, S-IHT and S-CoSamP give better results compared to the standard sparse inpainting OMP algorithm. This result in the evidence that the inpainting performance improves thanks to the selection of a few similar frames.

This principle may naturally extends to other algorithms, including those for an analysis model or for convex problem formulations, and to other inpainting settings like declipping.

References

[1] A. Adler, V. Emiya, M. G. Jafari, M. Elad, R. Gribonval, and M. Plumbley, "Audio inpainting," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 3, pp. 922–932, 2012.

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

[3] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2005.

[4] J. Mairal, F. R. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *IEEE International Conference on Computer Vision*, 2009.

[5] L. Li, J. Kan, and W. li, "Image denoising via robust simultaneous sparse coding," *Journal of Computers*, vol. 9, Jun. 2014.

[6] K. Siedenburg, M. Kowalski, and M. Dorfler, "Audio declipping with social sparsity," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1577–1578.

[7] E. Manilow and B. Pardo, "Leveraging repetition to do audio imputation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, oct 2017.

[8] Y. Bahat, Y. Y. Schechner, and M. Elad, "Self-content-based audio inpainting," *Signal Processing*, vol. 111, pp. 61–72, 2015.

[9] N. Perraudin, N. Holighaus, P. Majdak, and P. Balazs, "Inpainting of long audio segments with similarity graphs," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1083–1094, jun 2018.

[10] I. Toumi and V. Emiya, "Sparse non-local similarity modeling for audio inpainting," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Canada, 2018.

[11] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[12] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proc. Asilomar Conf. Signals, Systems, and Computers*, Nov. 1993.

[13] J. D. Blanchard, M. Cermak, D. Hanle, and Y. Jing, "Greedy algorithms for joint sparse recovery," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1694–1704, April 2014.

[14] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for Simultaneous Sparse Approximation: Part I: Greedy Pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572–588, 2006.

[15] J. A. Tropp, "Algorithms for Simultaneous Sparse Approximation. Part II: Convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589–602, 2006.

[16] D. Needell and J. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[17] A. Makhzani and S. Valaee, "Reconstruction of jointly sparse signals using iterative hard thresholding," in *2012 IEEE International Conference on Communications (ICC)*, June 2012.

[18] L. Belmerhnia, E. H. Djermoune, and D. Brie, "Greedy methods for simultaneous sparse approximation," in *European Signal Processing Conference (EUSIPCO)*, 2014.

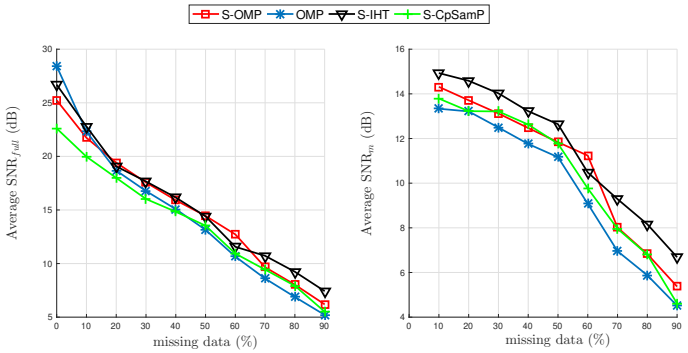

Fig. 7. Average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for single-channel speech sounds.

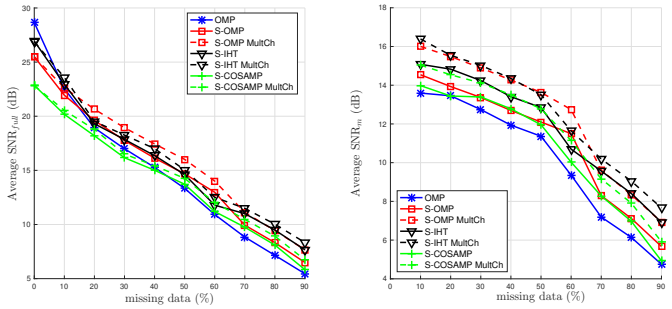

Fig. 10. Comparison of the average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for speech sounds in the multi channel case.

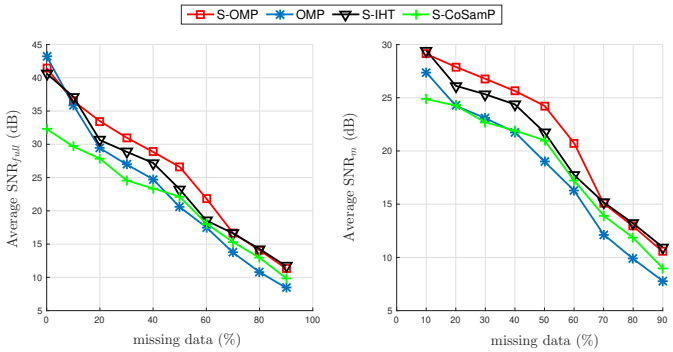

Fig. 8. Average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for single-channel music solo sounds.

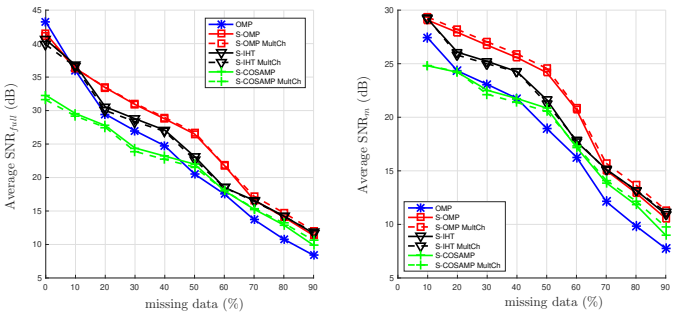

Fig. 11. Comparison of the average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for Music Solo sounds in the multi channel case.

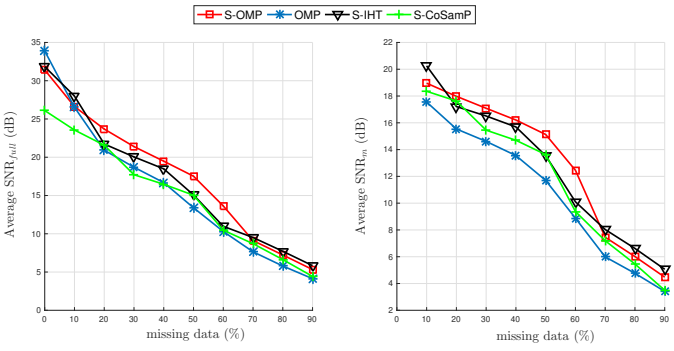

Fig. 9. Average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for single-channel music group sounds.

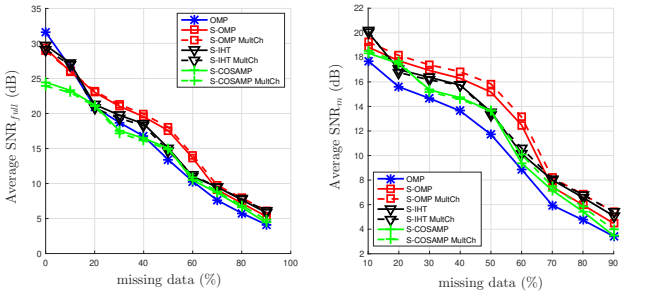

Fig. 12. Comparison of the average $\mathrm{SNR}_{\mathrm{full}}$ (left) and SNR_m (right) as a function of the percentage (p) of missing samples for Music Group sounds in the multi channel case.