Deep Prior-Based Audio Inpainting Using Multi-Resolution Harmonic Convolutional Neural Networks

Federico Miotello, *Graduate Student Member, IEEE*, Mirco Pezzoli, Luca Comanducci,
Fabio Antonacci, *Member, IEEE*, and Augusto Sarti, *Senior Member, IEEE*

***Abstract*—In this manuscript, we propose a novel method to perform audio inpainting, i.e., the restoration of audio signals presenting multiple missing parts. Audio inpainting can be interpreted in the context of inverse problems as the task of reconstructing an audio signal from its corrupted observation. For this reason, our method is based on a deep prior approach, a recently proposed technique that proved to be effective in the solution of many inverse problems, among which image inpainting. Deep prior allows one to consider the structure of a neural network as an implicit prior and to adopt it as a regularizer. Differently from the classical deep learning paradigm, deep prior performs a single-element training and thus it can be applied to corrupted audio signals independently from the available training data sets. In the context of audio inpainting, a network presenting relevant audio priors will possibly generate a restored version of an audio signal, only provided with its corrupted observation. Our method exploits a time-frequency representation of audio signals and makes use of a multi-resolution convolutional autoencoder, that has been enhanced to perform the harmonic convolution operation. Results show that the proposed technique is able to provide a coherent and meaningful reconstruction of the corrupted audio. It is also able to outperform the methods considered for comparison, in its domain of application.**

***Index Terms*—Audio inpainting, deep prior, deep learning.**

I. Introduction

THE advent of high-fidelity digital audio streaming and communications over the internet has significantly increased public awareness and expectations regarding sound quality across all kinds of recordings. It follows that there is a growing demand for the restoration of deteriorated or corrupted audio signals in order to enhance their perceived quality [1]. In fact, audio signals may often exhibit corrupted parts, in which multiple samples are absent, noisy, or unreliable. These alterations may be caused by several reasons, such as recording issues, packet loss during transmission, and faulty or aged storage supports. Among the different audio restoration tasks, we can cite for example audio denoising, which aims at reducing or removing unwanted background or acoustic noise [2] from audio recordings. Another common task is audio super resolution, also referred to as bandwidth extension [3]. The goal of audio super resolution is to generate high-quality audio from a down-sampled input that contains only a reduced number of the original samples [4]. The task of reconstructing an audio signal containing portions of missing samples, instead, is referred to as audio inpainting in the literature. This problem is closely related to audio Packet Loss Concealment (PLC) [5], [6], which can be seen as a subset applicative scenario of audio inpainting, when audio samples have been lost in the transmission over the internet. However, PLC has more computation time constraints, in order to enable the almost real-time and continuous fruition of audio data (e.g., in VoIP calls or music streaming). Moreover, typically PLC is defined as a causal process, meaning that the restoration algorithms only have visibility of the reliable packets preceding a missing portion [7], [8], [9]. On the other hand, audio inpainting is a more general problem, in which algorithms can exploit features coming both from before and after each missing portion of an audio signal. Additionally, time computation restrictions are typically not taken into account in this context.

In audio inpainting, the process of restoring lost information in audio signals is usually aimed at producing coherent and meaningful information while preventing the introduction of audible artifacts [10]. A precise reconstruction can be obtained only when the corrupted parts are a few milliseconds long. Instead, for corruptions in the order of hundreds of milliseconds or even seconds, an accurate estimate of the missing information might become unrealistic. Therefore, in order to provide a convincing reconstruction, inpainting methods need to produce a larger quantity of novel samples, that have to be semantically compatible with the surrounding context (i.e., the audio signal before and after each missing portion) [11].

Historically, the first methods tackled the audio inpainting problem by exploiting signal processing techniques. We can cite, for example, model-based techniques, such as auto-regressive models [12], [13], [14]. In this case, the goal is to infer the

Manuscript received 2 December 2022; revised 6 August 2023 and 22 September 2023; accepted 7 October 2023. Date of publication 16 October 2023; date of current version 30 October 2023. This work was supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of Next Generation EU, partnership on “Telecommunications of the Future” (PE00000001—Program “RESTART”). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Pejman Mowlaee. *(Corresponding author: Federico Miotello.)*

The authors are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milano, Italy (e-mail: federico.miotello@polimi.it; mirco.pezzoli@polimi.it; luca.comanducci@polimi.it; fabio.antonacci@polimi.it; augusto.sarti@polimi.it).

Digital Object Identifier 10.1109/TASLP.2023.3324556

parameters of the statistical process that generated the signal, only having access to its uncorrupted portions. The efficiency of these techniques relies on underlying assumptions on the characteristics of the considered audio signal (e.g., signal stationarity). For this reason, their performances rapidly degrade when these assumptions are no longer guaranteed [15].

Other signal processing methods, instead, rely on signal similarity to perform audio inpainting. In fact, speech, music, and other audio signals are composed of patterns and structures that occur multiple times. Information extracted from such repetitions demonstrated to be effective in the prediction of missing samples [16]. For example, in [7], the authors proposed a novel method to perform long audio inpainting that follows an optimization strategy carried out on similarity graphs. This data structure relates audio segment belonging to the corrupted signal according to time-frequency features and finally determines the most natural fit for the substitution of the lost content.

In recent years, given the results that deep learning achieved in many signal processing problems, audio inpainting has also been tackled by exploiting the power of deep neural networks [10], [11].The basic idea of deep learning-based audio inpainting is to train an artificial neural network to generate the missing data, given the uncorrupted parts of the signal. The model is provided only with indirect information about the content to generate, which however needs to be seamlessly inserted within the existing audio signal [11]. Nonetheless, these methods have shown to be effective in the inpainting of medium-length and long parts ($> 50\,\mathrm{ms}$). For example, in [11], the authors introduced GACELA (Generative Adversarial Context Encoder for Long Audio inpainting), an audio inpainting method based on a Generative Adversarial Network (GAN). GACELA proved to be able to restore missing musical audio data with a duration ranging between hundreds of milliseconds to a few seconds.

In general, the superior performances of deep learning techniques are motivated by their ability to learn realistic priors from a large quantity of training data. However, there are practical scenarios in which data sets are difficult to collect or even non-existing. In order to overcome this limitation, different techniques have been proposed [17], [18]. For example, in [17] the authors introduced Catch-A-Waveform (CAW), a GAN-based system in which the network is trained on a single audio signal analyzed at increasing sampling frequency. CAW is able to perform different audio restoration tasks, including audio inpainting. In this last case, the system is conditioned only on the available information surrounding the missing portions of the signal, producing reconstructions comparable to the ones provided by GACELA [11]. These approaches, in which a neural network is conditioned on a single data sample, can be interpreted in the context of deep prior. Deep prior [19] is a paradigm for the solution of inverse problems and has been initially proposed to address image restoration tasks. In the paper [19], the authors showed that, differently from the classical deep learning paradigm, the structure of a deep neural network is sufficient to capture most low-level statistics needed to solve such problems, prior to any learning on large data sets. Through this approach, a network learns how to map a random noise realization into the solution of the assigned task, relying on a single data item. In addition to the image domain, deep prior has also been successfully applied in the context of seismic data [20], room impulse responses [21] and vibrometric data [22].

In this article, we propose a method that exploits the deep prior paradigm to perform audio inpainting. In particular, we consider audio signals presenting multiple missing portions (also called gaps), each one in the order of tens of milliseconds. This kind of corruptions is relevant in the context of telecommunications, in which transmission errors (e.g., packet loss), might introduce several gaps of that size [23]. Moreover, this is a scale where the non-stationary characteristic of audio already becomes non-negligible [10], making the problem more challenging.

The proposed algorithm, that will be referred in the following as Deep Prior Audio Inpainting (DPAI), is based on the time-frequency representation of audio signals and exploits the MultiResUNet [24] architecture. Moreover, differently from previously proposed solutions, in our architecture we exploit the harmonic convolution operation [25]. Unlike regular convolution, this operation helps deep networks model low-level audio characteristics by aggregating information using a harmonics-shaped convolutional filter [25], thus leading to a better audio reconstruction. We compare DPAI with the Similarity Graph Algorithm (SGA) [7], and Catch-A-Waveform (CAW) [17]. Results show that our solution outperforms both reference methods in the inpainting of audio signals presenting multiple holes, each one ranging from $40\,\mathrm{ms}$ to $80\,\mathrm{ms}$. Therefore, the obtained performances demonstrate that the proposed approach can be effectively applied in the context of audio inpainting. The audio prior embedded in the network structure, provided by the combination of a multi-resolution approach and of the harmonic convolution, allows us to consistently reconstruct the missing parts of a corrupted audio signal.

The rest of this manuscript is organized as follows. In Section II we formalize the audio inpainting problem in the context of the deep prior paradigm. In Section III we present DPAI. In Section IV we show the results of the comparison between our method and the ones considered as reference. Finally, in Section V, we draw the conclusions of our work.

A demo web page for the project is available at the following link: `https://fmiotello.github.io/dpai`.

II. Problem Formulation

A. Signal Model

Let us consider a discrete audio signal $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]^T$, with $\mathbf{x} \in \mathbb{R}^{N\times 1}$, sampled at sampling frequency F_s. $\mathbf{X} \in \mathbb{C}^{M\times L}$ (an example is depicted in Fig. 1(a)) is the time-frequency representation of $\mathbf{x}$, with M frequency bins and L time frames. We denote by $\tilde{\mathbf{x}}$ a corrupted or incomplete observation of $\mathbf{x}$, and by $\tilde{\mathbf{X}} \in \mathbb{C}^{M\times L}$ (Fig. 1(b)), its time-frequency representation (i.e., STFT). Hence, some time frames of $\tilde{\mathbf{X}}$ are missing and we introduce a masking vector $\mathbf{s} \in \mathbb{R}^{1\times L}$, which indicates whether a time frame of $\tilde{\mathbf{X}}$ is lost (i.e., presents an overlap with the missing portions of the audio signal) or present,

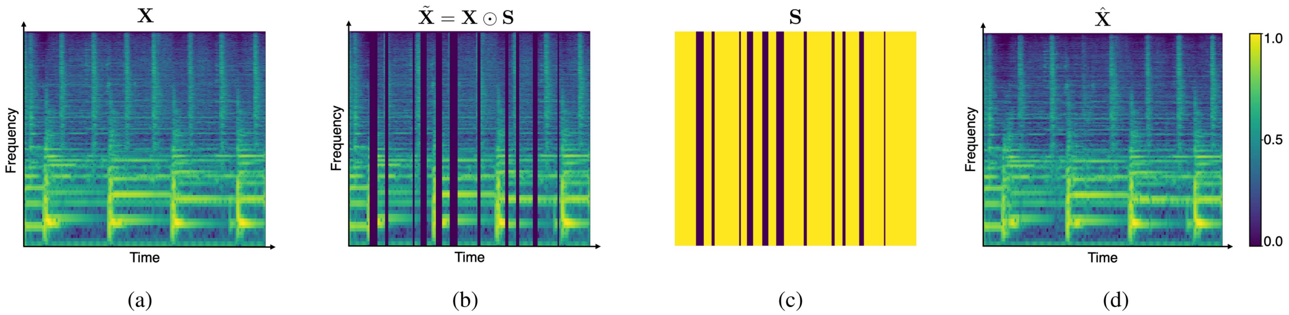


Fig. 1. Examples of (a) uncorrupted spectrogram, (b) corrupted spectrogram, (c) mask, and (d) reconstructed spectrogram.

as

$$\mathbf{s}_i = \begin{cases} 1, & \text{if the } i\text{th frame is present;} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

From $\mathbf{s}$ we can obtain $\mathbf{S} \in \mathbb{R}^{M \times L}$ (Fig. 1(c)), which is a time-frequency mask, as

$$\mathbf{S} = \mathbf{j}\mathbf{s} \tag{2}$$

where $\mathbf{j} \in \{1\}^{M \times 1}$ is an all-ones vector. A partial observation of $\mathbf{X}$ can then be defined as

$$\tilde{\mathbf{X}} = \mathbf{X} \odot \mathbf{S}, \tag{3}$$

where $\odot$ indicates the Hadamard product. In this context, we interpret audio inpainting in the framework of inverse problems. In practice, we aim at finding

$$\hat{\mathbf{X}} = F(\tilde{\mathbf{X}}), \quad \hat{\mathbf{X}} \approx \mathbf{X}, \tag{4}$$

which is an estimate of $\mathbf{X}$, starting from the available observation $\tilde{\mathbf{X}}$. F is a reconstruction operation that restores the missing portions of $\tilde{\mathbf{X}}$. Finally, from $\hat{\mathbf{X}}$ (Fig. 1(d)) it is possible to revert to the time-domain representation $\hat{\mathbf{x}}$ (i.e., the reconstructed audio signal) through the inverse STFT.

B. Deep Prior Approach

The solution to the inpainting problem requires the inversion of the relation in (3). This is an ill-posed inverse problem, whose solution $\hat{\mathbf{X}}^*$ is typically constrained in order to obtain meaningful results. In particular, it is necessary to add some kind of a priori information on the optimal solution. This is usually expressed in the form of a regularizer, that can be a handcrafted feature which models some implicit characteristic of the audio signal, such as sparsity [26], or can be learned from data, as it happens in deep learning audio inpainting methods [10], [11]. Formally, the solution can be expressed as an optimization problem in the form

$$\hat{\mathbf{X}}^* = \arg\min_{\hat{\mathbf{X}}} E\left(\hat{\mathbf{X}} \odot \mathbf{S}, \tilde{\mathbf{X}}\right) + R\left(\hat{\mathbf{X}}\right), \tag{5}$$

where $E(\cdot)$ is a data fidelity term (e.g., Mean Squared Error) that measures some distance between the reconstructed and observed data, and $R(\cdot)$ is the regularizer. It is worth noting that in (5) the distance between the reconstructed spectrogram and the corrupted one is computed ignoring the missing frames.

Inspired by solutions in the context of images [19], seismic data [20], room impulse responses [21] and vibrometric data [22], we propose to adopt a deep prior approach as an alternative regularization strategy for the solution of (5). Following this approach, let us consider a deep neural network as a generator described by the function f_θ, such as $\hat{\mathbf{X}} = f_\theta(\mathbf{Z})$, where θ is the set of trainable parameters of the considered network and $\mathbf{Z}$ is a random noise realization given as input to the network. Deep prior focuses on finding the parameters θ^*, that allow the considered neural network to estimate $\hat{\mathbf{X}}$ from $\mathbf{Z}$. In this framework the optimization problem in (5), can be reformulated as

$$\hat{\mathbf{X}}^* = f_{\theta^*}(\mathbf{Z}), \tag{6}$$

where

$$\theta^* = \arg\min_{\theta} E\left(f_\theta(\mathbf{Z}) \odot \mathbf{S}, \tilde{\mathbf{X}}\right). \tag{7}$$

Through this procedure, by minimizing (7) rather than (5), we search for the solution to the inverse problem in the space of the neural network parameters instead of the space of the model [20].

Also in (7), the data fidelity $E(\cdot)$ between the reconstructed spectrogram, given as output by the neural network, and the corrupted one, is computed ignoring the missing frames. However, unlike (5), in (7) the minimization is not constrained by an explicit regularization term but its role is replaced by the inherent a priori information provided by the network itself. Although the fit is performed on $\tilde{\mathbf{X}}$, the prior provided by the network enables to reconstruct also the missing parts of the spectrogram. It follows that the regularization ability of the neural network is linked to the structure of the architecture that drives the minimization - performed iteratively through gradient descent - towards solutions consistent with the prior [19].

It is also worth noting that, although $\hat{\mathbf{X}}^*$ is the output of an artificial neural network, our approach does not exploit the deep learning paradigm where a training phase is performed over an extensive data set of examples. In fact, only the corrupted observation $\tilde{\mathbf{X}}$ is used in the reconstruction process and the deep neural network implicitly assumes the role of prior information that exploits correlations in the data to learn its inner structure. For this reason, the choice of a specific architecture is crucial to find a suitable and coherent solution [19], [20].

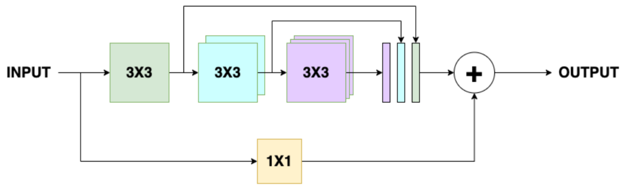


Fig. 2. MultiRes block: the outputs of the three consecutive convolutional layers are concatenated and then added to the residual connection.

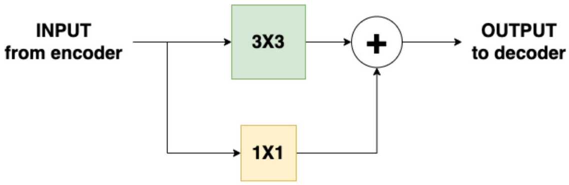


Fig. 3. Res Path block: the output of the convolutional layer is added to the residual connection.

III. Proposed Method

In this section we present DPAI, describing the architecture of the employed neural network and its fundamental characteristics. In particular, the considered network exploits a multi-resolution processing of data and it is enhanced with the harmonic convolution operation. Both these features, discussed in the following, proved to be essential in order to successfully model audio signals and thus achieve an effective regularization of (7).

A. Network Architecture

The proposed network architecture is based on MultiResUNet, a convolutional model proposed in [24] and characterized by an autoencoder structure. MultiResUNet has been introduced as an improvement of the well-known U-Net architecture [27], originally used for multimodal medical image segmentation, and exploits a multi-resolution approach to process data. In the context of audio signals, the multi-resolution processing of data allows the model to extract complex structures with dependencies at various frequency and temporal scales. This approach has been exploited by other audio inpainting techniques [11], [17], showing promising results.

In order to implement the multi-resolution processing of data, MultiResUNet replaces the standard convolutional layers with the so-called MultiRes blocks. Such blocks are inspired by the inception architecture [28], which analyzes the input features at different scales, using parallel convolutional filters with increasing kernel size. As a way to enhance this scheme, in [29], the authors show that convolutions with larger spatial filters (e.g., 5×5 or 7×7), can be factorized by using multiple layers of smaller convolutions (e.g., 3×3), that operate in series. In this way, for example, it is possible to replace a 5×5 convolutional block with two layers of 3×3 convolutional blocks. For this reason, in [24] the authors introduce the MultiRes Block, represented in Fig. 2, in which three 3×3 convolutional blocks operate in series and the output of each block is concatenated to the output of the following ones. Finally, the concatenated outputs are added to a residual connection, consisting in a 1×1 convolutional layer. As shown in [24], the result of the MultiRes block processing scheme is similar as if the processing was carried out using in parallel one 3×3 block, one 5×5 block and one 7×7 block, concatenating their outputs. As a result, this approach enables the multi-resolution processing proposed in [28], limiting the number of learnable parameters needed for the implementation.

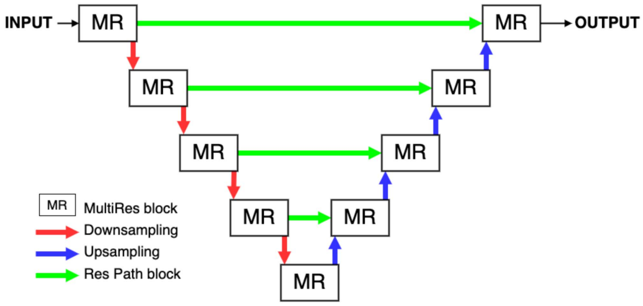
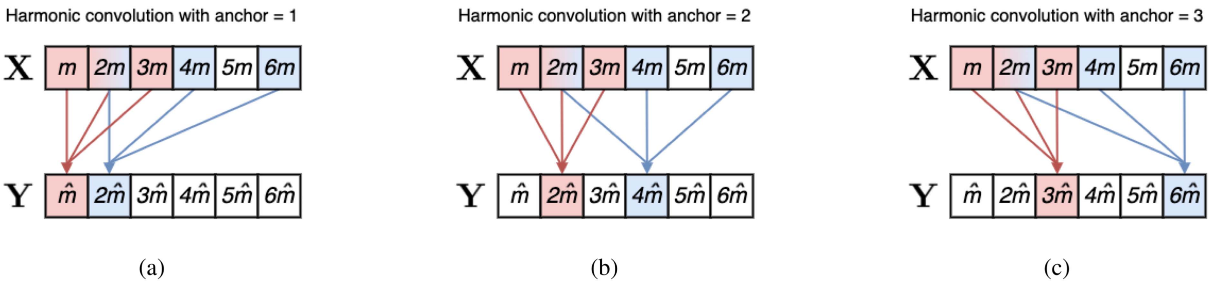


Fig. 4. Proposed MultiResUNet architecture.

Another important feature of MultiResUnet concerns the Res Path blocks (see Fig. 3). Res Path blocks replace classic skip connections, which are a key element of U-Net architectures [27] and enable the feature propagation between corresponding layers in encoder and decoder. Nonetheless, as explained in [24], skip connections are prone to generate the so-called semantic gap. In fact, features at the encoder that are passed unprocessed to the decoder are characterized by a low informative level, since they come from earlier stages of the processing. In the decoder, instead, features are derived from a higher-level representation of the bottleneck and possible previous decoding layers. The combination of these features, that often present a semantic discrepancy, can lead to a performance reduction. To address this semantic gap and alleviate the disparity between the encoder and decoder features, the authors of the MultiResUNet architecture [24] introduced convolutional layers along the shortcut connections, providing additional non-linear transformations to the encoder features before merging them with the ones in the decoder. The use of 3×3 filters helps account for the further processing done during decoding. In addition, the 1×1 filters in the residual connections allow the model to capture extra spatial information. This solution, as noted in [24], is able to reduce the semantic gap and thus improve the network performance.

In our implementation of MultiResUNet (see Fig. 4) we adopt 5 MultiRes blocks in the encoder with the corresponding decoding layers connected through Res Path blocks. The 3 filters in the MultiRes blocks contain 21, 43 and 64 kernels, respectively. The downsampling is performed through 3×3 convolutional layers with stride value of 2. On the other side, the upsampling is achieved with nearest-neighbor interpolation, using an upsampling factor equal to 2. Throughout the network, we adopted batch normalization and LeakyReLU as non-linear activation function after each convolutional layer.


Harmonic convolution with anchor = 1
X
m
2m
3m
4m
5m
6m
Y
m̂
2m̂
3m̂
4m̂
5m̂
6m̂
Harmonic convolution with anchor = 2
Harmonic convolution with anchor = 3

(a) (b) (c)

Fig. 5. Harmonic convolution (a) with anchor $n = 1$, the output frequency in $\mathbf{Y}$ is interpreted as the fundamental frequency of the convolution. In (b) the anchor is $n = 2$, thus the output frequency corresponds to the first harmonic. (c) Harmonic convolution with anchor $n = 3$.

Finally, in the practical implementation of the network, we treat complex STFT coefficients as two separate real-valued channels. This approach allows us to avoid the phase reconstruction step [30], [31] necessary for other common methods that operate on the magnitude only [10], [11].

B. Harmonic Convolution

In our implementation of MultiResUNet, convolutional layers in the encoder (MultiRes blocks, Res Path blocks, and downsampling layers), perform the so-called harmonic convolution [25], [32]. Harmonic convolution operates on the time-frequency representation of a given audio signal. While regular discrete 2D convolution in deep neural networks aggregates information on local regions of the input, harmonic convolution interprets the frequency dimension of the kernel as weights for a harmonic series at each target frequency location. This finds motivation in the structure of harmonic sounds, in which higher harmonics are located at integer multiples of a fundamental frequency, and which are also closely related to human sound perception. It has been demonstrated that harmonic convolution helps deep networks to model priors as an inductive bias in audio signals [25].

Formally, given the spectrogram $\mathbf{X}$ and a convolutional kernel $\mathbf{K}$, harmonic convolution is defined as a mapping from $\mathbf{X}[m, l]$ to $\mathbf{Y}[\hat{m}, \hat{l}]$, where

$$\mathbf{Y}[\hat{m}, \hat{l}] = \sum_{k=1}^{K_m} \sum_{l=1}^{K_l} \mathbf{X}\left[\frac{k\hat{m}}{n}, \hat{l} - l\right] \mathbf{K}[k, l] \,. \tag{8}$$

K_m and K_l are the kernel sizes along frequency and time dimensions respectively, $\hat{m}$ and $\hat{l}$ are the time and frequency output locations, and n an extra parameter called anchor. The output $\mathbf{Y}[\hat{m}, \hat{l}]$ is computed as a weighted sum of the input $\mathbf{X}$ at multiples of target frequency $\hat{m}$. Moreover, the anchor value n can be exploited to indicate the order of harmonics at location $\hat{m}$. For example, as represented in Fig. 5, using $n = 1$, $\hat{m}$ is interpreted as the fundamental frequency, using $n = 2$, $\hat{m}$ is interpreted as the first harmonic, and so on. In this way, the harmonic convolution is then computed considering $\frac{\hat{m}}{n}$ as the base frequency [32].

For the implementation of harmonic convolution we followed the method proposed in [32]. In the article, the authors present a faster calculation strategy for harmonic convolution, called Harmonic Lowering. As far as anchor values are concerned, we decided to vary them (shown in Table I) inside each convolutional block. We empirically verified that these values helped us obtain better results, since they allow us to exploit more harmonics other than the fundamental only.

TABLE I
Anchor Values Used in Each Convolutional Block

Block	Anchor value
MultiRes block	
Residual connection	1
Conv. layer #1	2
Conv. layer #2	3
Conv. layer #2	4
Res Path block	
Residual connection	1
Conv. layer #1	2
Downsampling	
Conv. layer #1	1

C. Loss Function

The training of the network is based on the minimization of the total reconstruction loss (7) between the output of the network and the observed spectrogram, computed only on the uncorrupted parts. This allows us to find the optimal weights θ^* of the network, needed to find the optimal solution (6). To this end, the loss function is computed as

$$\begin{aligned} \ell_{\text{tot}} = \alpha_1 [&\text{MSE}(\Re(\hat{\mathbf{X}}) \odot \mathbf{S}, \Re(\tilde{\mathbf{X}})) + \\ &+ \text{MSE}(\Im(\hat{\mathbf{X}}) \odot \mathbf{S}, \Im(\tilde{\mathbf{X}}))] + \alpha_2 \, \text{MSS}(\hat{\mathbf{X}} \odot \mathbf{S}, \tilde{\mathbf{X}}), \end{aligned} \tag{9}$$

where $\Re$ is the operator that retrieves the real part of a spectrogram and $\Im$ is the operator that retrieves the imaginary part of a spectrogram. In particular, (9) is the weighted sum of a Mean Squared Error (MSE) and a Multi-Scale Spectrogram (MSS) loss [33]. The MSE, between an observed spectrogram $\hat{\mathbf{X}}$ and its prediction $\tilde{\mathbf{X}}$, is computed as

$$\text{MSE}(\hat{\mathbf{X}}, \tilde{\mathbf{X}}) = \frac{1}{ML} \sum_{m=1}^{M} \sum_{l=1}^{L} (\hat{\mathbf{X}}[m, l] - \tilde{\mathbf{X}}[m, l])^2, \tag{10}$$

where M and L are the sizes of the spectrograms along frequency and time dimensions respectively.

TABLE II
STFT PARAMETERS USED TO COMPUTE MSS LOSS FUNCTION

FFT size	Window size	Hop length
2048	1200	240
1024	600	120
512	240	50

Note: all configurations use Hann window.
All values are expressed in samples, considering a sampling frequency F_s = 16 kHz.

The MSS loss instead, is composed of different spectrogram losses, computed changing each time the STFT analysis parameters (i.e., FFT size, window size, and hop length). The combination of multiple metrics, computed using different analysis parameters, helps the model learn time-frequency characteristics of the analyzed signal. Moreover, it also prevents the network from being biased by a fixed STFT representation, which may result in sub-optimal performances [33]. More specifically, our MSS loss is computed as the average of the total error at each of the P considered set of STFT parameters:

$$\begin{aligned}\mathrm{MSS}(\hat{\mathbf{X}} \odot \mathbf{S}, \tilde{\mathbf{X}}) = \frac{1}{P}\sum_{p=1}^{P} \ell_{\mathrm{sc}}(\hat{\mathbf{X}}_p, \tilde{\mathbf{X}}_p) \\ + \ell_{\log \mathrm{STFT}}(\hat{\mathbf{X}}_p, \tilde{\mathbf{X}}_p) + \ell_{\mathrm{STFT}}(\hat{\mathbf{X}}_p, \tilde{\mathbf{X}}_p) \\ + \ell_{\mathrm{phs}}(\hat{\mathbf{X}}_p, \tilde{\mathbf{X}}_p),\end{aligned} \tag{11}$$

where $\hat{\mathbf{X}}_p$ and $\tilde{\mathbf{X}}_p$ are estimated by first computing the inverse STFT of $\hat{\mathbf{X}} \odot \mathbf{S}$ and $\tilde{\mathbf{X}}$ respectively, and then applying the STFT again, using analysis parameters of scale p. In particular, the term ℓ_{sc} accounts for the spectral convergence between two spectrograms $\mathbf{X}$ and $\mathbf{Y}$, and is computed as

$$\ell_{\mathrm{sc}}(\mathbf{X}, \mathbf{Y}) = \frac{\| |\mathbf{X}| - |\mathbf{Y}| \|_F}{\| |\mathbf{Y}| \|_F}, \tag{12}$$

where $\| \cdot \|_F$ is the Frobenius norm. ℓ_{sc} emphasizes errors on spectral components presenting high energy. Conversely, $\ell_{\log \mathrm{STFT}}$ is defined as

$$\ell_{\log \mathrm{STFT}}(\mathbf{X}, \mathbf{Y}) = \frac{1}{N} \| \log(|\mathbf{X}|) - \log(|\mathbf{Y}|) \|_1, \tag{13}$$

where $\| \cdot \|_1$ is the ℓ_1 norm and N the number of STFT frames. It is exploited in order to improve the fit of low-amplitude components of the spectrogram [34]. The term ℓ_{STFT} is given by

$$\ell_{\mathrm{STFT}}(\mathbf{X}, \mathbf{Y}) = \frac{1}{N} \| |\mathbf{X}| - |\mathbf{Y}| \|_1, \tag{14}$$

and computes the linear STFT magnitude loss. Finally, the spectral phase loss accounts for the phase component reconstruction and is defined as

$$\ell_{\mathrm{phs}}(\mathbf{X}, \mathbf{Y}) = \mathrm{MSE}(\angle \mathbf{X}, \angle \mathbf{Y}). \tag{15}$$

The values α_1 and α_2 in (9) have been obtained through grid search and are set to $\alpha_1 = 1$ and $\alpha_2 = 0.1$. Following [33], we adopt the STFT parameters indicated in Table II to compute (11).

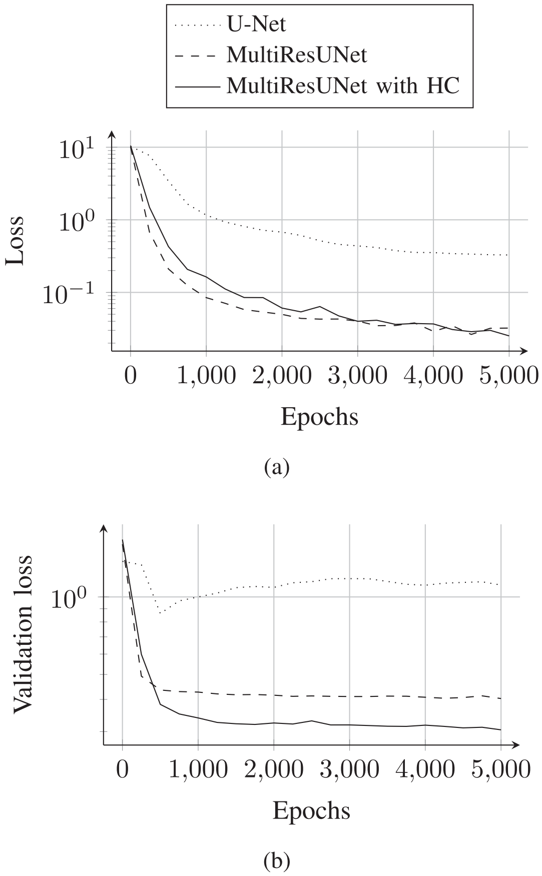


Fig. 6. Comparison of loss (a) and validation loss (b) evolution (in log scale) during the fitting of a corrupted spectrogram with three different models: U-Net, MultiResUNet and MultiResUNet with harmonic convolution.

IV. RESULTS

In this section we first present a preliminary experiment that justifies the network architecture chosen for DPAI. Then we evaluate the proposed method by performing both quantitative and perceptual tests. For each test, we compare the results obtained using DPAI to the ones achieved by CAW [17] and SGA [7].

A. Network Analysis

In order to validate the network architecture adopted in DPAI, we performed a preliminary experiment that compares the loss curves obtained during the training of our model on a single data sample, with the ones produced by two different models presenting a comparable number of parameters. In particular, we compared the MultiResUNet model enhanced with harmonic convolution (ours) to the MultiResUNet [24] and the standard U-Net architecture [27]. Both MultiResUNet architectures (with and without harmonic convolution) contain 2015252 parameters, while the considered U-Net architecture has 2158578 parameters.

In Fig. 6(a) we show the loss evolution during the fitting of a corrupted spectrogram following the deep prior approach and exploiting the loss function presented in Section III-C.

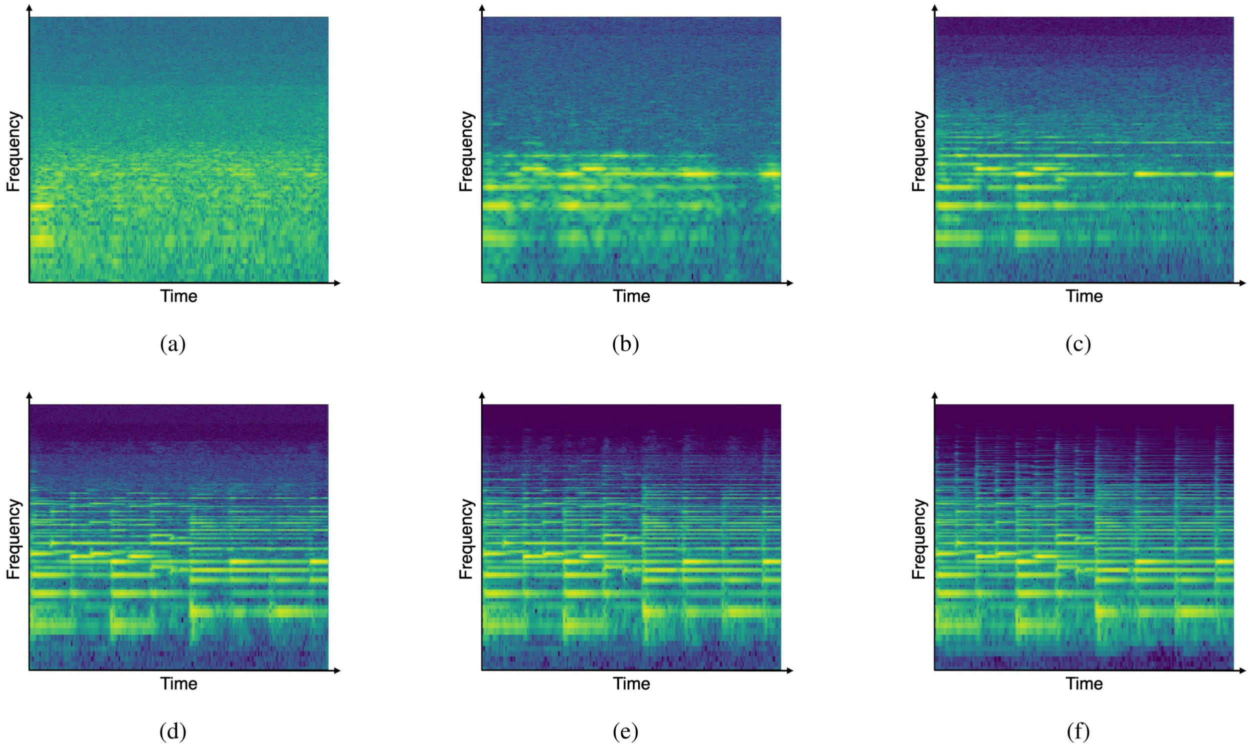


Fig. 7. Reconstruction of a corrupted spectrogram at increasing epochs: (a) 30 epochs, (b) 50 epochs, (c) 150 epochs, (d) 500 epochs, (e) 1000 epochs, and (f) 5000 epochs.

Both MultiResUNet with and without convolution obtain similar curves, reaching quite rapidly small values of the loss, while the U-Net model, instead, stabilizes on much higher values.

Fig. 6(b), instead, compares the evolution of the validation loss, namely, computed only on the missing portions of the spectrogram. Hence, it is possible to examine the reconstruction capabilities of the considered models. The reconstruction ability of the U-Net architecture, on the missing portions of the spectrogram, is limited, as shown by the validation loss value that diverges as the epochs increase. MultiResUNet, instead, reaches quite quickly small values of validation loss, and even smaller values when the harmonic convolution is employed. Both results demonstrate that the use of a multi-resolution approach and the exploitation of the harmonic convolution allow the chosen model to extract strong audio priors from the corrupted spectrogram, thus leading to a better reconstruction. This process of reconstruction is illustrated in Fig. 7. In particular, it shows the output of the MultiResUNet model with harmonic convolution at different iterations. First, starting from a noise input, only some blurry structures are recovered. Then, the higher energy elements, usually located at the lower frequencies, start to be reconstructed. Finally, also low energy components are gradually reconstructed as the number of epochs increases, simultaneously achieving the generation of the missing frames.

B. Dataset

We tested the performance of the proposed audio inpainting technique on a corpus of 15 audio signals, coming from piano [35], popular music [36] and speech datasets [37]. The length of each audio signal is $5\,\mathrm{s}$, with a sampling frequency $F_s = 16\,\mathrm{kHz}$. Similarly to the approach proposed in [38], we considered in the experiments 5 different values of cumulative gaps duration, namely $200\,\mathrm{ms}$, $400\,\mathrm{ms}$, $600\,\mathrm{ms}$, $800\,\mathrm{ms}$ and $1\,\mathrm{s}$. Given a cumulative gaps duration, we generated a masking vector, as defined in (1), for each audio signal, encoding multiple gaps with random positions and random lengths ranging from $40\,\mathrm{ms}$ to $80\,\mathrm{ms}$. The corrupted observations are obtained through (3).

C. Objective Metrics

To numerically evaluate the three methods under comparison, we used the Normalized Mean Squared Error (NMSE) averaged on the whole data set, i.e.

$$\mathrm{NMSE} = 10\log_{10}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2^2}{\|\mathbf{x}_i\|_2^2}\right), \tag{16}$$

where N is the cardinality of the data set and $\|\cdot\|_2$ is the ℓ_2 norm. In particular, we computed NMSE in two different configurations: $\mathrm{NMSE}_{\mathrm{tot}}$ and $\mathrm{NMSE}_{\mathrm{miss}}$. In the first case we consider the whole reconstructed signal $\hat{\mathbf{x}}$ and therefore, also the uncorrupted parts of the audio impact on NMSE. On the contrary, $\mathrm{NMSE}_{\mathrm{miss}}$ is evaluated only on the portions of signal that were missing in the observation.

Moreover, for the speech samples of our dataset, we also considered the Perceptual Evaluation of Speech Quality (PESQ) metric [39], that is a widely used method to assess speech quality in terms of Mean Opinion Score (MOS) [40]. Although PESQ was originally developed to model subjective tests in the context of telecommunications, it is also used to evaluate the performances of speech enhancement algorithms [41], [42], [43]. We computed PESQ for corrupted and reconstructed signals and then we computed the difference between these two values, in order to assess the quality improvement (or deterioration) provided by the considered reconstruction methods. The results were then averaged on the whole data set.

D. Subjective Tests

In order to evaluate the perceived signal quality, we also conducted a human auditory test, comparing the reconstructions obtained via DPAI with those generated by CAW [17] and SGA [7]. Specifically, similarly to the test proposed in [17], for each question participants listened to a corrupted audio signal, along with three possible reconstructions (one for each method), and were asked to rate the plausibility of each one. We considered a reconstruction plausible if every gap had been filled with information coherent with its surroundings and if it did not contain audible artifacts. The entire test consisted of 10 questions, with each audio signal being 5 s long.

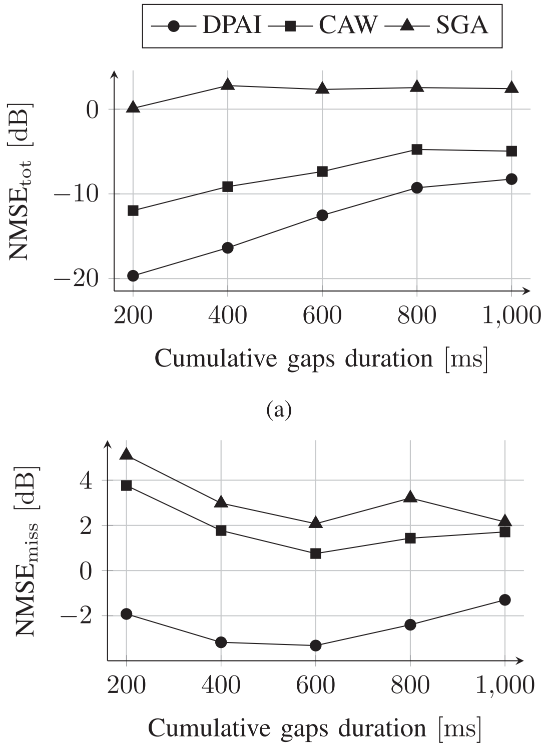
E. Setup

DPAI results were computed training the model proposed in Section III for 5000 epochs, using the Adam optimizer [44] and learning rate set to 0.01. The network's input consists of a uniform noise tensor with variance equal to 0.1. In particular, from a preliminary analysis of the network we found out that uniformly distributed noise consistently produced better results. Thus, we decided to use it for all subsequent experiments. Moreover, similarly to [19], in order to reinforce convergence at each iteration the input tensor is perturbed with additional zero-mean white noise of 0.03 variance.

By default settings, SGA is set to produce results with a difference in length from the input corrupted signal. This trade-off is aimed at achieving reconstructions that present smoother transitions with the original signal. However, this approach prevents a direct numerical comparison of signals with different lengths in terms of NMSE and PESQ. To address this limitation, adjusting SGA parameters, we forced the length of the reconstructed signal to match the length of the initial one.

CAW exploits the processing of audio signals at progressively increasing sampling rates to train the neural network. Specifically, for computing the results using CAW, we adopted 6 different sampling frequencies: 1000 Hz, 1600 Hz, 2500 Hz, 8000 Hz, 12000 Hz and 16000 Hz. We derived the adopted scales based on a set of preliminary analyses that demonstrated reduced computational time while not significantly affecting the reconstruction error with respect to the scales proposed in the original paper [17].

It is important to underline that both methods used for comparison, preserve, by design, the uncorrupted parts of the input signal, inserting the reconstructions inside the gaps. DPAI, instead, has the capability of reconstructing the whole signal. In fact, similarly other deep-prior-based methods [19], [20], [21], we replicate in output also the uncorrupted parts of the input.



(a)

(b)

Fig. 8. Quantitative test results. Total NMSE (a) and NMSE evaluated only on the portions of signal that were missing (b).

As far as SGA is concerned, the method does not provide newly generated signals, neither for the missing portions nor for the uncorrupted ones. Instead, SGA seamlessly inserts sections of the uncorrupted signal into the gaps, selecting the ones that ensure smooth integration with the surrounding signal context, based on the similarity criteria [7]. In contrast, CAW generates a completely new signal by using its GAN structure, based only on the uncorrupted segments of the signal. Nevertheless, the output is crafted to sound "in the style of" the original signal, without attempting to exactly replicate it, even in the uncorrupted sections. Consequently, as proposed in [17] the inpainting is performed seamlessly blending the reconstructed segments into the original corrupted audio.

F. Discussion

The quantitative results in terms of NMSE are shown in Fig. 8. We can observe that the proposed solution is able to outperform both reference methods, suggesting that the output signal produced by DPAI reconstructs more accurately the original audio signal. In particular, the gain in NMSE attained by DPAI with respect to CAW and SGA is at least $-3.3\,\mathrm{dB}$ and $-3\,\mathrm{dB}$ in terms $\mathrm{NMSE}_{\mathrm{tot}}$ and $\mathrm{NMSE}_{\mathrm{miss}}$, respectively, as visible inspecting Fig. 8(a) and Fig. 8(b).

SGA is the method that on average performs the worst, as it is not able to return a coherent reconstruction. Our interpretation is

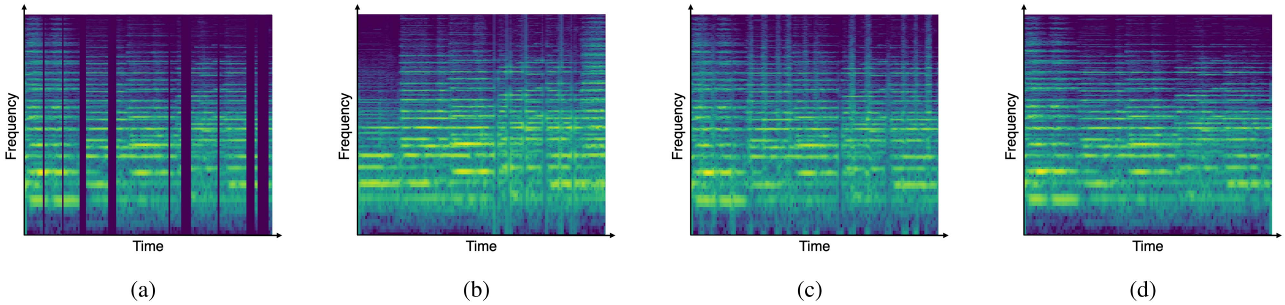


Fig. 9. (a) Corrupted spectrogram, (b) SGA reconstruction, (c) CAW reconstruction, and (d) DPAI reconstruction.

that the algorithm needs more uncorrupted information to build an effective similarity graph. For this reason, for the computation of the results, we provided SGA with 5 s of additional reliable signal before and after each audio. With a more reliable context the algorithm possibly identifies more repetitions and recurring structures in the audio and thus has more options to choose from for the substitution. However, given the results, this is not sufficient, since the reconstructed signals are significantly distant from the original ones, in terms of NMSE. Moreover, in some cases, the algorithm also modifies uncorrupted portions of the audio signal, overlapping them with non-contextual information.

Reconstructions computed using CAW, instead, seem to be more reliable than the ones provided by SGA. However, the transitions between the reconstructed portions and the original ones are evident, and the reconstructed portions sound noisier than the original ones.

The plot in Fig. 8(a) shows that as the cumulative duration of the gaps increases, a degradation of reconstruction performances of the methods is observed. Instead, as represented in the plot in Fig. 8(b), the reconstruction capabilities of the methods remain almost constant, if only the reconstructed portions are considered. These observations are particularly true for DPAI and CAW, and suggest that the degradation of performances of inpainting methods is particularly affected by a larger number of gaps. This is justified by the fact that the presence of more gaps leads to the introduction of more discontinuities and artifacts that highly affect $\mathrm{NMSE}_{\mathrm{tot}}$.

The differences mentioned above can also be noticed by inspecting Fig. 9, which represents the spectrograms of a corrupted audio signal, in Fig. 9(a), and the three corresponding reconstructions, obtained using SGA, in Fig. 9(b), CAW, in Fig. 9(c) and DPAI in Fig. 9(d). It can be noticed how SGA fails to obtain a coherent reconstruction, also affecting the uncorrupted portions of the audio signal. Nonetheless, it does not insert too many visible artifacts. CAW, instead, is able to reconstruct most of the missing portions. However, as it can be seen in the spectrogram, the reconstructions are noisy and show a discontinuity with the surrounding context. As a result, these corruptions can be perceived as artifacts during the playback of the restored audio signal. On the contrary, the reconstruction provided by DPAI appears to be the most accurate: it restores the missing information in all the gaps and does not present any visible discontinuity. The transition between reconstructions and uncorrupted parts is smooth and not perceivable during playback. Only when the audio signal is corrupted by longer gaps, the signal energy appears to drop within the restored portions, making them noticeable.

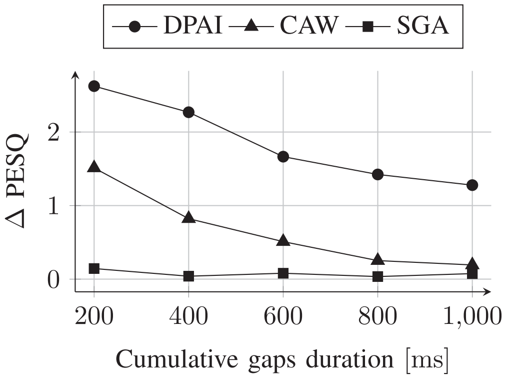


Fig. 10. Perceptual test results. PESQ difference between corrupted and reconstructed speech signals.

TABLE III
Results of the Perceptual Test

DPAI	CAW	SGA
69 %	10 %	21 %

Fig. 10 shows the mean PESQ increment between corrupted and reconstructed speech audio signals. In particular, we computed the increment as

$$\Delta\mathrm{PESQ} = \mathrm{PESQ}(\mathbf{x}, \hat{\mathbf{x}}) - \mathrm{PESQ}(\mathbf{x}, \tilde{\mathbf{x}}), \tag{17}$$

where PESQ is the function that computes the PESQ value between a reference and a degraded audio signal. We can observe that DPAI is able to obtain a better increment than CAW and SGA. This proves that speech signals reconstructed with our method are characterized by higher intelligibility and quality, making it suitable for speech restoration tasks.

Table III shows the results of the perceptual test. The test was carried out on 11 subjects, ranging from age 24, to age 36. All the subjects were asked to use consumer-grade headphones or higher while taking the test, in order to better assess the audio quality of the reconstructions. 69% of the participants chose DPAI as the method that provided the most plausible reconstruction. SGA was instead chosen in the 21% of the cases, and CAW in the 10% of the cases. As far as SGA and CAW are concerned, the results of the perceptual test show an opposing trend to the numerical tests, in which CAW obtained better results. This outcome is justified by the fact that while SGA also affects the uncorrupted parts of the audio signals, the algorithm does not introduce many artifacts or discontinuities, since the portions of signal used to fill the gaps are extracted from the input signal itself. Audio signals computed using CAW instead, while achieving better results in terms of quantitative metrics, present many discontinuities and audible artifacts in proximity of the transition between reconstructions and uncorrupted parts.

The subjective test confirms the results of the quantitative tests and proves that the proposed method is able to obtain coherent and meaningful reconstructions of corrupted audio signals, in the considered domain of application.

V. Conclusion

In this work, we presented a novel audio inpainting method, based on a deep prior approach. The method is able to restore an audio signal affected by multiple gaps, with a duration ranging between $40\,\mathrm{ms}$ and $80\,\mathrm{ms}$. Unlike classic deep learning techniques, in which the models are trained on large data sets, our solution only relies on the corrupted audio observation it aims to reconstruct. It is based on the MultiResUNet architecture [24] and enhanced with the use of harmonic convolutions [25]. Both quantitative and perceptual tests show that, in its domain of application, the proposed solution outperforms the methods considered for comparison. Nonetheless, DPAI also presents some downsides. In particular, as the size of individual gaps increases, the performances quickly degrade: the overall reconstruction gets worse and the gaps more noticeable as the signal energy drops within the filled portions. Moreover, the single-element training approach, while it does not need to exploit a large data set, is limited by the time necessary to process a single corrupted audio signal. In order to overcome this limitation it could be possible to exploit pre-trained architectures, that have already been conditioned on different audio signals.

Further developments for this method could include the extraction, from the corrupted audio, of explicit features such as beat and chord-tracking, since their exploitation could improve the inpainting quality and introduce complex structures and dependencies.

References

[1] S. Godsill, P. Rayner, and O. Cappé, *Digital Audio Restoration.* Berlin, Germany: Springer, 2002.

[2] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, “Speech enhancement and dereverberation with diffusion-based generative models,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 31, pp. 2351–2364, 2023.

[3] E. Moliner, J. Lehtinen, and V. Välimäki, “Solving audio inverse problems with a diffusion model,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2023, pp. 1–5.

[4] V. Kuleshov, S. Z. Enam, and S. Ermon, “Audio super-resolution using neural nets,” in *Proc. Int. Conf. Learn. Representations*, 2017. [Online]. Available: https://dblp.org/rec/conf/iclr/KuleshovEE17.html?view=bibtex

[5] Y. Guan, G. Yu, A. Li, C. Zheng, and J. Wang, “TMGAN-PLC: Audio packet loss concealment using temporal memory generative adversarial network,” in *Proc. Interspeech*, 2022, pp. 565–569.

[6] L. Ou and Y. Chen, “Concealing audio packet loss using frequency-consistent generative adversarial networks,” in *Proc. IEEE 5th Int. Conf. Pattern Recognit. Artif. Intell.*, 2022, pp. 826–831.

[7] N. Perraudin, N. Holighaus, P. Majdak, and P. Balazs, “Inpainting of long audio segments with similarity graphs,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 6, pp. 1083–1094, Jun. 2018.

[8] L. Diener, S. Sootla, S. Branets, A. Saabas, R. Aichner, and R. Cutler, “Interspeech 2022 audio deep packet loss concealment challenge,” in *Proc. Interspeech*, 2022, pp. 850–584.

[9] J.-M. Valin et al., “Real-time packet loss concealment with mixed generative and predictive model,” in *Proc. Interspeech*, 2022, pp. 570–574.

[10] A. Marafioti, N. Perraudin, N. Holighaus, and P. Majdak, “A context encoder for audio inpainting,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 27, no. 12, pp. 2362–2372, Dec. 2019.

[11] A. Marafioti, P. Majdak, N. Holighaus, and N. Perraudin, “GACELA: A generative adversarial context encoder for long audio inpainting of music,” *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 1, pp. 120–131, Jan. 2021.

[12] P. A. Esquef, V. Välimäki, K. Roth, and I. Kauppinen, “Interpolation of long gaps in audio signals using the warped burg’s method,” in *Proc. 6th Int. Conf. Digit. Audio Effects*, 2003, pp. 8–11.

[13] I. Kauppinen and K. Roth, “Audio signal extrapolation–theory and applications,” in *Proc. DAFx*, 2002, pp. 105–110.

[14] I. Kauppinen and J. Kauppinen, “Reconstruction method for missing or damaged long portions in audio signal,” *J. Audio Eng. Soc.*, vol. 50, no. 7/8, pp. 594–602, 2002.

[15] O. Mokrỳ and P. Rajmic, “Audio inpainting: Revisited and reweighted,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2906–2918, 2020.

[16] I. Toumi and V. Emiya, “Sparse non-local similarity modeling for audio inpainting,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2018, pp. 576–580.

[17] G. Greshler, T. Shaham, and T. Michaeli, “Catch-a-waveform: Learning to generate audio from a single short example,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 20916–20928.

[18] A. Turetzky, T. Michelson, Y. Adi, and S. Peleg, “Deep audio waveform prior,” in *Proc. Interspeech*, 2022, pp. 2938–2942.

[19] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Deep image prior,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9446–9454.

[20] F. Kong, F. Picetti, V. Lipari, P. Bestagini, X. Tang, and S. Tubaro, “Deep prior-based unsupervised reconstruction of irregularly sampled seismic data,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 7501305.

[21] M. Pezzoli, D. Perini, A. Bernardini, F. Borra, F. Antonacci, and A. Sarti, “Deep prior approach for room impulse response reconstruction,” *Sensors*, vol. 22, no. 7, 2022, Art. no. 2710.

[22] R. Malvermi, F. Antonacci, A. Sarti, and R. Corradi, “Prediction of missing frequency response functions through deep image prior,” in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2021, pp. 106–110.

[23] Y. Bahat, Y. Y. Schechner, and M. Elad, “Self-content-based audio inpainting,” *Signal Process.*, vol. 111, pp. 61–72, 2015.

[24] N. Ibtehaz and M. S. Rahman, “MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation,” *Neural Netw.*, vol. 121, pp. 74–87, 2020.

[25] Z. Zhang et al., “Deep audio priors emerge from harmonic convolutional networks,” in *Proc. Intl. Conf. Learn. Representations*, 2020. [Online]. Available: https://dblp.org/rec/conf/iclr/ZhangWG0T0F20.html?view=bibtex

[26] L. Stanković and M. Brajović, “Analysis of the reconstruction of sparse signals in the DCT domain applied to audio signals,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 26, no. 7, pp. 1220–1235, Jul. 2018.

[27] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*, 2015, pp. 234–241.

[28] C. Szegedy et al., “Going deeper with convolutions,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.

[30] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, “Deep Griffin–Lim iteration: Trainable iterative phase reconstruction using neural network,” *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 1, pp. 37–50, Jan. 2021.

[31] Z. Pruša and P. L. Søndergaard, “Real-time spectrogram inversion using phase gradient heap integration,” in *Proc. Int. Conf. Digit. Audio Effects*, 2016, pp. 17–21.

[32] H. Takeuchi, K. Kashino, Y. Ohishi, and H. Saruwatari, “Harmonic lowering for accelerating harmonic convolution for audio signals.,” in *Proc. INTERSPEECH*, 2020, pp. 185–189.

[33] R. Yamamoto, E. Song, and J.-M. Kim, “Parallel waveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 6199–6203.

[34] S. Ö. Arık, H. Jun, and G. Diamos, “Fast spectrogram inversion using multi-head convolutional neural networks,” *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 94–98, Jan. 2019.

[35] C. Hawthorne et al., “Enabling factorized piano music modeling and generation with the MAESTRO dataset,” in *Proc. Int. Conf. Learn. Representations*, 2019.

[36] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, “FMA: A dataset for music analysis,” in *Proc. 18th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2017.

[37] D. Snyder, G. Chen, and D. Povey, “MUSAN: A music, speech, and noise corpus,” 2015, *arXiv:1510.08484v1*.

[38] G. Morrone, D. Michelsanti, Z.-H. Tan, and J. Jensen, “Audio-visual speech inpainting with deep learning,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 6653–6657.

[39] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, “Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (Cat. No 01CH37221)*, 2001, pp. 749–752.

[40] *Vocabulary for Performance, Quality of Service and Quality of Experience*, Recommendation ITU-T P.10/G.100, Int. Telecommun. Union Telecommun. Standardization Sector, Geneva, Switzerland, 2017. [Online]. Available: https://www.itu.int/rec/T-REC-P.10-201711-I/en

[41] S.-W. Fu, T.-y. Hu, Y. Tsao, and X. Lu, “Complex spectrogram enhancement by convolutional neural network with multi-metrics learning,” in *Proc. IEEE 27th Int. Workshop Mach. Learn. Signal Process.*, 2017, pp. 1–6.

[42] K. Tan and D. Wang, “Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 380–390, 2020.

[43] Y. Xia, S. Braun, C. K. A. Reddy, H. Dubey, R. Cutler, and I. Tashev, “Weighted speech distortion losses for neural-network-based real-time speech enhancement,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 871–875.

[44] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. 3rd Int. Conf. Learn. Representations {ICLR}*, San Diego, CA, USA, May 7–9, 2015.

Federico Miotello (Graduate Student Member, IEEE) was born in Gallarate, Italy, in 1995. He received the bachelor’s degree in computer science from the University of Milan, Milan, Italy, in 2019, and the master’s degree in music and acoustic engineering from Politecnico di Milano, Milan, in 2022. In 2022, he joined the Image and Sound Processing Lab, Politecnico di Milano as a Ph.D. student. His research interests include arrays of audio transducers and audio signal processing techniques based on deep learning.

Mirco Pezzoli received the M.S. degree (*cum laude*) in computer engineering and the Ph.D. degree in information engineering from the Politecnico di Milano, Milan, Italy, in 2017 and 2021, respectively. After two years as a Postdoctoral Researcher, he joined the Department of Electronics, Information and Bioengineering, Politecnico di Milano as junior Assistant Professor. His research interests include multichannel audio signal processing, sound field reconstruction, and musical acoustics.

Luca Comanducci was born in Genova, Italy, in 1991. He received the B.Sc. degree in music information science from the University of Milan, Milan, Italy, in 2014, the M.Sc. in computer science and engineering, and the Ph.D. degree in information technology from Politecnico di Milano, Milan, Italy, in 2018 and 2022, respectively. He is currently working as a Postdoc with Politecnico di Milano. His research interests include application of deep learning techniques to spatial audio problems, networked music performance, and music information retrieval.

Fabio Antonacci (Member, IEEE) was born in Bari, Italy, in 1979. He received the Laurea degree in telecommunication engineering and the Ph.D. degree in information engineering from Politecnico di Milano, Milan, Italy, in 2004 and 2008, respectively. He is currently an Associate Professor with Politecnico di Milano. He is author of more than 130 articles in proceedings of international conferences and peer-reviewed journals. His research interests include musical acoustics, in particular on the development of innovative non invasive measurement methodologies, space-time processing of audio signals, for both speaker and microphone arrays (source localization, acoustic scene analysis, and rendering of spatial sound) and on modeling of acoustic propagation (visibility-based beam tracing).

Augusto Sarti (Senior Member, IEEE) received the Ph.D. degree in information engineering from the University of Padova, Padua, Italy, in 1993, with a joint graduate program with the University of California, Berkeley. In 1993, he joined the Politecnico di Milano, Milan, Italy, where he is currently a Full Professor. From 2013 to 2017, he held professorship with the University of California, Davis, CA, USA. At PoliMI, he currently coordinates the research activities of the Musical Acoustics Lab and Sound and Music Computing Lab, and the M.Sci. Program in music and acoustic engineering. He has coauthored more than 300 scientific publications on international journals and congresses and numerous patents in the multimedia signal processing area. His research interests include the area of audio and acoustic signal processing, with particular focus on audio and acoustic signal processing, music information retrieval, and musical acoustics. He served two terms with the IEEE Technical Committee on Audio and Acoustics Signal Processing. He was an Associate Editor for IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, and the Senior Area Editor of IEEE SIGNAL PROCESSING LETTERS. He is also with the EURASIP board of directors.

Open Access funding provided by ‘Politecnico di Milano’ within the CRUI CARE Agreement