

---

# DeepWave: A Recurrent Neural-Network for Real-Time Acoustic Imaging

---

**Matthieu Simeoni** \*

IBM Zurich Research Laboratory  
meo@zurich.ibm.com

**Sepand Kashani** †

École Polytechnique Fédérale de Lausanne (EPFL)  
sepand.kashani@epfl.ch

**Paul Hurley**

Western Sydney University  
paul.hurley@westernsydney.edu.au

**Martin Vetterli**

École Polytechnique Fédérale de Lausanne (EPFL)  
martin.vetterli@epfl.ch

## Abstract

We propose a recurrent neural-network for real-time reconstruction of acoustic camera spherical maps. The network, dubbed DeepWave, is both physically and algorithmically motivated: its recurrent architecture mimics iterative solvers from convex optimisation, and its parsimonious parametrisation is based on the natural structure of acoustic imaging problems. Each network layer applies successive filtering, biasing and activation steps to its input, which can be interpreted as generalised deblurring and sparsification steps. To comply with the irregular geometry of spherical maps, filtering operations are implemented efficiently by means of graph signal processing techniques. Unlike commonly-used imaging network architectures, DeepWave is moreover capable of directly processing the complex-valued raw microphone correlations, learning how to optimally back-project these into a spherical map. We propose moreover a smart physically-inspired initialisation scheme that attains much faster training and higher performance than random initialisation. Our real-data experiments show DeepWave has similar computational speed to the state-of-the-art delay-and-sum imager with vastly superior resolution. While developed primarily for acoustic cameras, DeepWave could easily be adapted to neighbouring signal processing fields, such as radio astronomy, radar and sonar.

## 1 Introduction

**Motivation** An *acoustic camera (AC)* [26, 8, 18, 24] is a multi-modal imaging device that allows one to visualise in real-time sound emissions from every *direction* in space. This is typically achieved by overlaying on the live video from an optical camera a heatmap representing the intensity of the ambient directional sound field, recovered from the simultaneous recordings of a *microphone array* [3, 42]. Most commercial acoustic cameras recover the sound intensity field by combining linearly the correlated microphone recordings with a *Delay-And-Sum (DAS) beamformer* [42, Chapter 5]. The beamformer acts as an *angular filter* [20, 21], steering sequentially the array sensitivity pattern –or *beamshape*– towards various directions where the sound intensity field is probed. Acoustic images obtained this way are cheap to compute, but are *blurred* by the beamshape of the microphone array,

---

\*Corresponding author. Matthieu Simeoni is also affiliated to the École Polytechnique Fédérale de Lausanne (EPFL), with email address `matthieu.simeoni@epfl.ch`

†Matthieu Simeoni and Sepand Kashani have contributed equally to this work. Sepand Kashani was in part supported by the Swiss National Science Foundation grant number 200021 181978/1, “SESAM - Sensing and Sampling: Theory and Algorithms”.

and hence exhibit poor angular resolution [49, 7, 48]. The severity of this blur can be shown [53] to be proportional to the ratio  $\lambda/D$ , where  $D$  is the diameter of the microphone array and  $\lambda$  the sound wavelength. Because of the relatively large wavelengths of acoustic waves in the audible range, this blur can be significant in practice: a 30 cm diameter microphone array has an angular resolution at 5 kHz (an  $E\flat$ ) of approximately 10 degrees, against  $7 \cdot 10^{-4}$  degrees for a standard optical camera at 790 THz (violet). Moreover, acoustic cameras are often deployed in confined environments [34], requiring them to be as *compact* and *portable* as possible, which limits<sup>3</sup> further the achievable angular resolution.

The advent of *compressed sensing* techniques [14, 44] –and their wide adoption in imaging sciences [54, 4, 32]– have inspired algorithmic solutions [48, 7, 11, 12] to the acoustic imaging problem, promising vastly improved angular resolutions. Unfortunately, these methods proved ill-suited for real-time purposes. Indeed, they often rely on iterative solvers, such as *proximal gradient descent* (PGD) [37] or its accelerated variants [2, 31]. While exhibiting a fast convergence rate [2], such methods still require on the order of a few dozen iterations to converge in practice, making them unable to cope with the high refresh-rate<sup>4</sup> of acoustic cameras. For this reason, and despite their clear superiority in terms of resolving power, nonlinear imaging methods have not yet replaced the suboptimal DAS imager in the software stack of commercial acoustic cameras.

The recent eruption of *deep learning* [33, 56, 10] in the field of imaging sciences may however seal the fate of DAS for good. Indeed, this new imaging paradigm leverages *neural-networks* [28] to reduce dramatically the image formation time. Unlike compressed-sensing methods which proceed iteratively, neural networks *encode* the image reconstruction process in a cascade of linear and nonlinear transformations *trained* on a very large number of input/output example pairs. Once properly trained, a neural-network can be efficiently evaluated for some input data to produce images of high quality, with similar accuracy and resolution as state-of-the-art compressed-sensing methods [33]. Network architectures used for inverse imaging [22, 17, 56, 10, 40] are most often *convolutional neural-networks* (CNNs), directly adapted from generic architectures developed for *image classification* and *segmentation* [45]. While suitable for image processing tasks such as *denoising*, *super-resolution* or *deblurring* [38, 6], such architectures are ill-suited [33] for more complex image reconstruction problems where the input data may not consist of an image, as is the case in *biomedical imagery* [4, 32], *interferometry* [54] or acoustic imaging. Moreover, and particularly limiting for our current purposes, standard convolutional architectures cannot handle images with non-Euclidean domains [13] such as *spherical maps* [41] produced by omnidirectional acoustic or optical cameras.

To overcome these limitations, *recurrent* architectures [16, 50, 30, 33] have been proposed, by *unrolling* iterative convex optimisation algorithms. Such networks are not only able to handle non-image inputs, but also have greater interpretability than generic CNNs. For example, Gregor and LeCun proposed in their pioneering work [16] a *recurrent neural-network* (RNN) dubbed LISTA<sup>5</sup>, inspired from the popular *iterative soft-thresholding algorithm* (ISTA)[2].<sup>6</sup> Their network can be seen as generalising ISTA, allowing for the normally fixed gradient and proximal steps occurring at each iteration of the algorithm to be learnt from the data: update steps of ISTA are replaced by a cascade of recurrent layers with trainable parameters. The depth of the resulting RNN is typically much smaller than the number of iterations required for ISTA to converge. Roughly speaking, the network is learning *shortcuts* in the reconstruction space, allowing it to achieve a prescribed reconstruction accuracy faster than gradient-based iterative methods.<sup>7</sup>

While the effectiveness of LISTA was verified on small images from the MNIST dataset (784 pixels) [16], its application to large-scale imaging problems remains challenging. This is mainly due to the huge number of weights parametrising the network which, in the fully-connected case, grows as the number of pixels to the square. Storing<sup>8</sup> –let alone learning– all those weights quickly becomes intractable for increasing resolutions. As a potential fix, Gregor and LeCun recommended sparsifying the network by *pruning* layer connections. While they showed that such a pruning could reduce the number of parameters in the network by 80% without affecting too much the performance of the

<sup>3</sup>Remember that the blur spread is inversely proportional to the microphone array diameter.

<sup>4</sup>An acoustic camera typically updates the acoustic image a dozen times per second.

<sup>5</sup>LISTA stands for learned iterative soft-thresholding algorithm.

<sup>6</sup>ISTA is an instance of proximal gradient descent for *penalised basis pursuit* problems [52].

<sup>7</sup>Of course, such shortcuts will most likely only be valid for the distribution of inputs and outputs implicitly defined by the training set, which should hence be carefully crafted for the network to generalise well in practice.

<sup>8</sup>For a 1 megapixel image, the weights parametrising the network would be approximately 8 Gb in size.

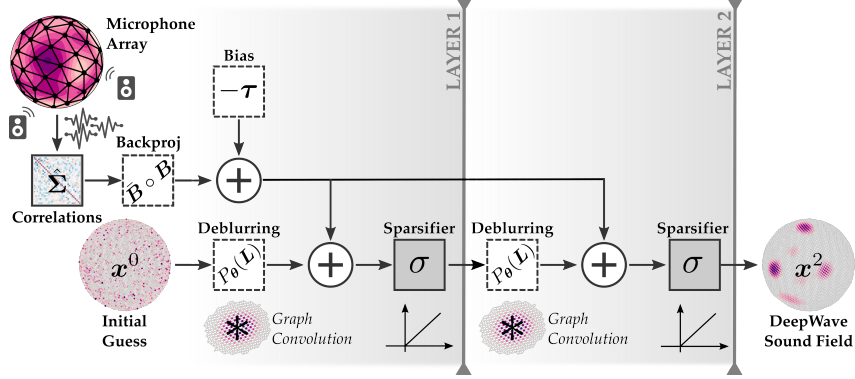


Figure 1: DeepWave’s recurrent architecture (1) for  $L = 2$  layers and random initialisation. Learnable parameters of the network are denoted by dashed boxes. Affine operations are denoted by white boxes and nonlinear activations by grey boxes.

latter, this is still insufficient for large-scale problems, and additional structure must be considered on network layers. Such structure is however often very dependent on the problem at hand.

**Contributions** In this work, we propose the first realistic architecture of a LISTA neural-network adapted to acoustic imaging. Our custom architecture, dubbed *DeepWave*, is capable of rendering high-resolution spherical maps of real-life sound intensity fields in milliseconds. DeepWave is tailored to the acoustic imaging problem, leveraging fully its underlying structure so as to minimise the number of network parameters. The latter is easy to train, with a typical training time of less than an hour on a general-purpose CPU. Unlike most state-of-the-art neural-network architectures, it moreover readily supports *complex*-valued input vectors, making it capable of directly processing the raw correlated microphone recordings. Assuming a microphone array with  $M$  microphones, the instantaneous covariance matrix  $\hat{\Sigma} \in \mathbb{C}^{M \times M}$  of the microphone recordings is processed by the network as follows (see also fig. 1):

$$\mathbf{x}^l = \sigma \left( P_{\theta}(\mathbf{L}) \mathbf{x}^{l-1} + [\overline{\mathbf{B}} \circ \mathbf{B}]^H \text{vec}(\hat{\Sigma}) - \boldsymbol{\tau} \right), \quad l = 1, \dots, L, \quad (1)$$

where  $\text{vec} : \mathbb{C}^{M \times M} \rightarrow \mathbb{C}^{M^2}$  is the *vectorisation operator* and  $\circ$  denotes the *Khatri-Rao product* (see appendix A<sup>9</sup> for definitions). The neurons  $\{\mathbf{x}^1, \dots, \mathbf{x}^L\} \subset \mathbb{R}_+^N$  at the output of each layer  $l$  of the depth  $L$  neural-network correspond to the acoustic image as it is processed by the network, with  $N$  the number of pixels. The neuron  $\mathbf{x}^0 \in \mathbb{R}_+^N$  defines the initial state of the network. The nonlinear *activation function*<sup>10</sup>  $\sigma : \mathbb{R} \rightarrow \mathbb{R}$  induces *sparsity* in the acoustic image, and is inspired by the proximal operator of an elastic-net penalty [37]. The remaining quantities, namely  $P_{\theta}(\mathbf{L})$ ,  $\mathbf{B}$  and  $\boldsymbol{\tau}$  are trainable parameters of the network, with various roles:

- *Deblurring*: the matrix  $P_{\theta}(\mathbf{L}) := \sum_{k=0}^K \theta_k \mathbf{L}^k \in \mathbb{R}^{N \times N}$  can be interpreted as a *deblurring matrix*, cleaning potential artefacts from the array beamshape. Following the approach of [41], it is defined as a polynomial of the *graph Laplacian*  $\mathbf{L} \in \mathbb{R}^{N \times N}$  based on the *connectivity graph* of the spherical tessellation in use, with learnable coefficients  $\boldsymbol{\theta} = [\theta_0, \dots, \theta_K] \in \mathbb{R}^{K+1}$ . Such parametrisation permits notably the interpretation of  $P_{\theta}(\mathbf{L})$  as a finite-support filter defined on the tessellation graph. Moreover, fast graph convolution algorithms are available for such filters [13].
- *Back-projection*: the operation  $[\overline{\mathbf{B}} \circ \mathbf{B}]^H \text{vec}(\hat{\Sigma}) = \text{diag}(\mathbf{B}^H \hat{\Sigma} \mathbf{B})$  (A.8) is a *back-projection*, mapping the raw microphone correlations to the image domain. Thanks to the convenient Khatri-Rao structure, this linear operation depends only on the matrix  $\mathbf{B} \in \mathbb{C}^{M \times N}$ .

<sup>9</sup>In all that follows, labels prefixed with roman letters refer to elements of the supplementary material.

<sup>10</sup>Typified by a rectilinear unit.

- *Bias*: the vector  $\boldsymbol{\tau} \in \mathbb{R}^N$  is a non-uniform *bias*, boosting or shrinking the neurons of the network. Since only positive neurons are activated by the nonlinearity  $\sigma$ , this biasing operation helps sparsify the final acoustic image.

The total number of learnable coefficients in DeepWave is *linear* in the number of pixels. The rationale behind DeepWave’s architecture is detailed in section 2, with theoretical justifications for the structures of the deblurring and back-projection linear operators. In section 3, we discuss network training, including initialisation and regularisation. We moreover derive the *forward-* and *backward-propagation* recursions<sup>11</sup> for our custom architecture, required for forming gradient steps. Finally, we test the architecture on synthetic as well as real data acquired with the *Pyramic array* [5, 46]. DeepWave is shown to have similar resolving power as state-of-the-art compressed-sensing methods, with a computational overhead similar to the DAS imager. To our knowledge, this is the first time a nonlinear imager of the kind achieves real-time performance on a standard computing platform. While developed primarily for acoustic cameras, DeepWave can easily be applied in neighbouring array signal processing fields [27], including radio astronomy, radar and sonar technologies.

## 2 Network architecture

In this section, we proceed similarly to [16, 50, 30] and construct DeepWave by studying the update equations of an iterative solver, namely proximal gradient descent applied to acoustic imaging.

### 2.1 Proximal gradient descent for acoustic imaging

In all that follows, we model the sound intensity field as a discrete *spherical map* with resolution  $N$ , specified by an *intensity vector*  $\mathbf{x} \in \mathbb{R}_+^N$  and a *tessellation*  $\Theta = \{\mathbf{r}_1, \dots, \mathbf{r}_N\} \subset \mathbb{S}^2$ . Spherical tessellations [19, 15] can be viewed as pixelation schemes for spherical geometries (see appendix B.1). As is customary in compressed-sensing, we propose to recover the sound intensity map by solving a convex optimisation problem (see appendix C):

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{R}_+^N} \frac{1}{2} \left\| \hat{\boldsymbol{\Sigma}} - \mathbf{A} \text{diag}(\mathbf{x}) \mathbf{A}^H \right\|_F^2 + \lambda [\gamma \|\mathbf{x}\|_1 + (1 - \gamma) \|\mathbf{x}\|_2^2], \quad (2)$$

where  $\|\cdot\|_F$  denotes the *Frobenius norm*,  $\gamma \in ]0, 1[$  and  $\lambda > 0$  are hyperparameters, and  $\hat{\boldsymbol{\Sigma}} \in \mathbb{C}^{M \times M}$  is the empirical covariance matrix of the microphone recordings. In a far-field context, the *forward map*  $\mathbf{A} \in \mathbb{C}^{M \times N}$  –linking the intensity vector to the microphone recordings– is commonly modelled by the so-called *steering matrix* [27]:  $[\mathbf{A}]_{mn} := \exp(-2\pi j \langle \mathbf{p}_m, \mathbf{r}_n \rangle / \lambda_0)$ , where  $\{\mathbf{p}_1, \dots, \mathbf{p}_M\} \subset \mathbb{R}^3$  are the microphone locations and  $\lambda_0 > 0$  the sound wavelength. Using properties (A.5) and (A.6) of the *vectorisation operator* and the Frobenius norm [23, 53], problem (2) can be re-written in vectorised form as:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{R}_+^N} \frac{1}{2} \left\| \text{vec}(\hat{\boldsymbol{\Sigma}}) - (\overline{\mathbf{A}} \circ \mathbf{A}) \mathbf{x} \right\|_2^2 + \lambda [\gamma \|\mathbf{x}\|_1 + (1 - \gamma) \|\mathbf{x}\|_2^2], \quad (3)$$

where  $\circ$  denotes the *Khatri-Rao product* (see definition A.3). Problem (3) is an *elastic-net penalised least-squares problem* [57], which seeks an optimal<sup>12</sup> trade-off between *data-fidelity* and *group-sparsity*. Group-sparsity is in this context better suited than traditional sparsity since acoustic sources are often diffuse. It is worth noting that, since the elastic-net functional is strictly convex for  $\gamma \in [0, 1[$ , problem (3) admits a unique solution. The latter can moreover be approximated by means of *proximal gradient descent (PGD)* [2], whose update equations are given here by (see appendix D):

$$\mathbf{x}^k = \text{ReLu} \left( \frac{\mathbf{x}^{k-1} - \alpha (\overline{\mathbf{A}} \circ \mathbf{A})^H \left[ (\overline{\mathbf{A}} \circ \mathbf{A}) \mathbf{x}^{k-1} - \text{vec}(\hat{\boldsymbol{\Sigma}}) \right] - \lambda \alpha \gamma}{2\lambda \alpha (1 - \gamma) + 1} \right), \quad k \geq 1, \quad (4)$$

where  $\mathbf{x}^0 \in \mathbb{R}^N$  is arbitrary,  $\alpha \leq 1 / \|\overline{\mathbf{A}} \circ \mathbf{A}\|_2^2$  is the *step size* and  $\text{ReLu}(x) := \max(x, 0)$  is the *rectified linear unit* [29], applied element-wise to a real vector.<sup>13</sup> The sequence of iterates  $\{\mathbf{x}^k\}_{k \in \mathbb{N}}$

<sup>11</sup>DeepWave implementation can be found on <https://github.com/imagingofthings/DeepWave>.

<sup>12</sup>The notion of optimality is defined here by the penalty parameter  $\lambda$ .

<sup>13</sup>Note that with  $\mathbf{x}^0 \in \mathbb{R}^N$ , every gradient step produces a real vector.

defined in (4) reduces the objective function in (3) at a rate  $O(1/k)$  [2]. Accelerated variants of proximal gradient descent have been proposed [2], which modify (4) with an extra *momentum term*:

$$\begin{cases} \mathbf{y}^k &= \text{ReLU} \left( \frac{\mathbf{x}^{k-1} - \alpha (\overline{\mathbf{A}} \circ \mathbf{A})^H \left[ (\overline{\mathbf{A}} \circ \mathbf{A}) \mathbf{x}^{k-1} - \text{vec}(\hat{\Sigma}) \right] - \lambda \alpha \gamma}{2\lambda\alpha(1-\gamma) + 1} \right), \quad k \geq 1, \\ \mathbf{x}^k &= \mathbf{y}^k + \omega^k (\mathbf{y}^k - \mathbf{y}^{k-1}) \end{cases} \quad (5)$$

where the *momentum sequence*  $\{\omega^k\}_{k \in \mathbb{N}}$  can be designed in various ways [31, 9]. In our experiments, we will use (5) as a baseline for speed comparisons, where  $\omega^k$  is updated according to Chambolle and Dossal’s strategy [9]:  $\omega^k = (k-1)/(k+d)$ ,  $k \geq 0$ , with  $d = 50$  [31]. The *accelerated proximal gradient descent (APGD)* method thus obtained is the fastest reported in the literature, with convergence rate  $o(1/k^2)$  [31]. Finally, we leverage the formulae  $(\overline{\mathbf{A}} \circ \mathbf{A}) \mathbf{x} = \text{vec}(\mathbf{A} \text{diag}(\mathbf{x})\mathbf{A}^H)$  (A.5), and  $(\overline{\mathbf{A}} \circ \mathbf{A})^H \text{vec}(\mathbf{R}) = \text{diag}(\mathbf{A}^H \mathbf{R} \mathbf{A})$  (A.8), to compute gradient steps efficiently in (5).

## 2.2 DeepWave : a PGD-inspired RNN for fast acoustic imaging

In practice PGD is terminated according to some stopping criterion. The intensity map  $\mathbf{x}^L$  obtained after  $L$  iterations of (4) can then be seen as the output of an RNN with depth  $L$  and intermediate neurons linked by the recursion formula:

$$\mathbf{x}^l = \text{ReLU} \left( \mathcal{D} \mathbf{x}^{l-1} + \mathcal{B} \text{vec}(\hat{\Sigma}) - \boldsymbol{\tau} \right), \quad l = 1, \dots, L. \quad (6)$$

We call this RNN the *oracle RNN*, since its weights  $\mathcal{D} \in \mathbb{R}^{N \times N}$ ,  $\mathcal{B} \in \mathbb{C}^{N \times M^2}$  and  $\boldsymbol{\tau} \in \mathbb{R}^N$  are not learnt but simply *given* to us by identifying (6) with (4):

$$\mathcal{D} = \frac{1}{\beta} \left[ I - \alpha (\overline{\mathbf{A}} \circ \mathbf{A})^H (\overline{\mathbf{A}} \circ \mathbf{A}) \right], \quad \mathcal{B} = \frac{\alpha}{\beta} (\overline{\mathbf{A}} \circ \mathbf{A})^H, \quad \boldsymbol{\tau} = \frac{\lambda \alpha \gamma}{\beta} \mathbf{1}_N, \quad (7)$$

where  $\beta = 2\lambda\alpha(1-\gamma) + 1$ . An analysis of (7) allows us moreover to interpret physically the affine operations performed by the oracle RNN. The matrix  $\mathcal{B}$  first is a *back-projection* operator, mapping the vectorised correlation matrix into a spherical map by applying the adjoint of the forward operator used in (3). The resulting spherical map is called a *dirty map*, and is equivalent to the DAS image [53, Section 5.2][55]. The matrix  $\mathcal{D}$  then is a *deblurring operator*, which subtracts at each iteration a fraction of the array beamshape from the spherical map, hence *cleaning* the latter of blur artefacts. The vector  $\boldsymbol{\tau}$  finally is an affine *shrinkage operator*, which biases uniformly the spherical map. The latter permits –in conjunction with the rectified linear unit– the *sparsification* of the spherical map and hence improve its angular resolution.

Since the oracle RNN is merely a reinterpretation of PGD, it inherits all its properties. In particular, it is capable of solving (3) with high accuracy for arbitrary input correlation matrices. Unfortunately, this great generalisability is typically obtained at the price of a very large number<sup>14</sup> of layers  $L$ , resulting in impractical reconstruction times. If one is however willing to sacrifice some of this generalisability, it is possible to reduce drastically the network depth by unfreezing the weights  $\mathcal{D}$ ,  $\mathcal{B}$ ,  $\boldsymbol{\tau}$  in (6), and allowing them to be *learnt* for some specific input distribution. This idea was first explored in the context of sparse coding by Gregor and LeCun [16], resulting in the LISTA network. A fully-connected architecture, corresponding to unconstrained  $\mathcal{D}$ ,  $\mathcal{B}$  and  $\boldsymbol{\tau}$ , would however result in  $\mathcal{O}(N^2)$  weights to be learnt, which is unfeasible in large-scale acoustic imaging problems. To overcome this issue, we propose in the next paragraphs a parsimonious parametrisation of  $\mathcal{D}$  and  $\mathcal{B}$ . The resulting RNN architecture, dubbed DeepWave, is given in (1) and depicted in fig. 1.

**Parametrisation of  $\mathcal{D}$**  Our parametrisation of  $\mathcal{D}$  is motivated by the following result, characterising the oracle deblurring kernel for *spherical microphone arrays*[42] (see proof in appendix E).

**Proposition 1.** *Consider a spherical microphone array, with diameter  $D$  and microphone directions  $\{\hat{\mathbf{p}}_1, \dots, \hat{\mathbf{p}}_M\} \subset \mathbb{S}^2$ , forming a near-regular tessellation of the sphere. Then, we have*

$$\left[ I - \alpha (\overline{\mathbf{A}} \circ \mathbf{A})^H (\overline{\mathbf{A}} \circ \mathbf{A}) \right]_{ij} \simeq \left[ \delta_{ij} - \alpha M^2 \text{sinc}^2 \left( \frac{D}{\lambda_0} \|\mathbf{r}_i - \mathbf{r}_j\| \right) \right], \quad \forall i, j \in \{1, \dots, N\} \quad (8)$$

<sup>14</sup>Even with momentum acceleration, PGD typically requires more than 50 iterations to converge. The oracle RNN obtained by unrolling PGD will consequently be very deep.

Algorithm 1 DeepWave forward propagation	Algorithm 2 DeepWave backward propagation
1: <b>Input:</b> $\hat{\Sigma}_t, \mathbf{x}_t^0, \hat{\mathbf{x}}_t, \boldsymbol{\theta}, \mathbf{B}, \boldsymbol{\tau}, \sigma$ 2: <b>Output:</b> $\mathcal{L}_t \in \mathbb{R}_+, \{\mathbf{s}_t^l\}_{l=1,\dots,L} \subset \mathbb{R}^N$ 3: 4: $\mathbf{y}_t \leftarrow \text{diag}(\mathbf{B}^H \hat{\Sigma}_t \mathbf{B}) - \boldsymbol{\tau}$ 5: <b>for</b> $l$ <b>in</b> $[1, \dots, L]$ <b>do</b> 6: $\mathbf{s}_t^l \leftarrow P_{\boldsymbol{\theta}}(\mathbf{L}) \mathbf{x}_t^{l-1} + \mathbf{y}_t$ 7: $\mathbf{x}_t^l \leftarrow \sigma(\mathbf{s}_t^l)$ 8: $\mathcal{L}_t \leftarrow \frac{1}{2} \ \hat{\mathbf{x}}_t - \mathbf{x}_t^L\ _2^2 / \ \hat{\mathbf{x}}_t\ _2^2$	1: <b>Input:</b> $\hat{\Sigma}_t, \mathbf{x}_t^0, \hat{\mathbf{x}}_t, \boldsymbol{\theta}, \mathbf{B}, \sigma, \{\mathbf{s}_t^l\}_{l=1,\dots,L}$ 2: <b>Output:</b> $\partial \boldsymbol{\theta} \in \mathbb{R}^{K+1}, \partial \mathbf{B} \in \mathbb{C}^{M \times N}, \partial \boldsymbol{\tau} \in \mathbb{R}^N$ 3: $(\partial \mathbf{x}, \partial \boldsymbol{\theta}, \partial \boldsymbol{\tau}) \leftarrow ((\sigma(\mathbf{s}_t^L) - \hat{\mathbf{x}}_t) / \ \hat{\mathbf{x}}_t\ _2^2, \mathbf{0}, \mathbf{0})$ 4: <b>for</b> $l$ <b>in</b> $[L, \dots, 1]$ <b>do</b> 5: $\partial \mathbf{s} \leftarrow \text{diag}(\sigma'(\mathbf{s}_t^l)) \partial \mathbf{x}$ 6: $\partial \mathbf{x} \leftarrow P_{\boldsymbol{\theta}}(\mathbf{L}) \partial \mathbf{s}$ 7: $\partial \boldsymbol{\tau} \leftarrow \partial \boldsymbol{\tau} - \partial \mathbf{s}$ 8: $[\partial \boldsymbol{\theta}]_k \leftarrow [\partial \boldsymbol{\theta}]_k + \partial \mathbf{s}^T T_k(\mathbf{L}) \sigma(\mathbf{s}_t^{l-1})$ 9: $\partial \mathbf{B} \leftarrow -2 \hat{\Sigma}_t \mathbf{B} \text{diag}(\partial \boldsymbol{\tau})$

Figure 2: Forward and backward algorithms to compute gradients of  $\mathcal{L}_t$  with respect to  $\boldsymbol{\theta}, \mathbf{B}, \boldsymbol{\tau}$ . For notational simplicity we use the shorthand  $\partial \boldsymbol{\alpha} = \partial \mathcal{L}_t / \partial \boldsymbol{\alpha}$ , and assume  $\sigma(\mathbf{s}_t^0) = \mathbf{x}_t^0$ .

where  $\lambda_0$  is the wavelength,  $\delta_{ij}$  denotes the Kronecker delta and  $\text{sinc}(x) := \sin(\pi x) / \pi x$  is the cardinal sine. Moreover, the approximation (8) is extremely good for  $M \geq 3 \lfloor \frac{2\pi D}{\lambda_0} \rfloor^2$ .

Proposition 1 tells us that, for spherical arrays with sufficient number of microphones<sup>15</sup>, the oracle deblurring operator  $\mathcal{D}$  in (7) corresponds actually to a sampled *zonal kernel* [35]:  $[\mathcal{D}]_{ij} = \kappa(\|\mathbf{r}_i - \mathbf{r}_j\|)$  for some  $\kappa: \mathbb{R}_+ \rightarrow \mathbb{R}$ . Since zonal kernels are used to define spherical convolutions [35],  $\mathcal{D}$  can hence be seen as a *discrete convolution operator* over the tessellation in use  $\Theta = \{\mathbf{r}_1, \dots, \mathbf{r}_N\}$ . Its bandwidth is moreover essentially finite, since coefficients  $[\mathcal{D}]_{ij}$  decay as  $1/\|\mathbf{r}_i - \mathbf{r}_j\|^2$ . As discussed in [41, 13], discrete spherical convolution operators with finite scope can be efficiently represented and implemented by means of *graph signal processing* [47] techniques. This leads us to consider the following parametrisation (see appendix B.3 for details):  $\mathcal{D} = P_{\boldsymbol{\theta}}(\mathbf{L}) := \sum_{k=0}^K \theta_k \mathbf{L}^k$ , where  $\boldsymbol{\theta} = [\theta_0, \dots, \theta_K] \in \mathbb{R}^{K+1}$ ,  $K$  controls the scope of the discrete convolution and  $\mathbf{L} \in \mathbb{R}^{N \times N}$  is the *Laplacian* [47] associated to the convex-hull graph of  $\Theta$ . Note that with this parametrisation, the number of parameters characterising  $\mathcal{D}$  drops from  $N^2$  to  $K+1$ , with  $K \ll N$ .

**Parametrisation of  $\mathcal{B}$**  The oracle back-projection operator (7) admits a factorisation in terms of the Khatri-Rao product. We decide hence to equip  $\mathcal{B}$  with a similar structure:  $\mathcal{B} = (\bar{\mathbf{B}} \circ \mathbf{B})^H$  for some learnable matrix  $\mathbf{B} \in \mathbb{C}^{M \times N}$ . With such a parametrisation, the number of parameters characterising  $\mathcal{B}$  drops from  $NM^2$  to  $NM$ . The Khatri-Rao structure guarantees moreover real-valued –and hence physically-interpretable– dirty maps.

### 3 Network training

To facilitate the description of the training procedure, we adopt the following shorthand notations.

- DeepWave( $\boldsymbol{\Omega}, L$ ) denotes a specific instance of the DeepWave network (1) with parameters  $\boldsymbol{\Omega} := \{\boldsymbol{\theta}, \mathbf{B}, \boldsymbol{\tau}\}$  and depth  $L$ .
- APGD( $\alpha, \lambda, \gamma$ ) denotes an instance of APGD (5), with tuning parameters  $(\alpha, \lambda, \gamma) \in \mathbb{R}_+^3$ .

The network parameters are chosen as minimisers of the following optimisation problem:

$$\hat{\boldsymbol{\Omega}} \in \arg \min_{\substack{\boldsymbol{\theta} \in \mathbb{R}^{K+1} \\ \mathbf{B} \in \mathbb{C}^{M \times N} \\ \boldsymbol{\tau} \in \mathbb{R}^N}} \frac{1}{T} \sum_{t=1}^T \underbrace{\frac{\|\hat{\mathbf{x}}_t - \mathbf{x}_t^L(\boldsymbol{\Omega})\|_2^2}{2 \|\hat{\mathbf{x}}_t\|_2^2}}_{:= \mathcal{L}_t} + \underbrace{\frac{\lambda_{\boldsymbol{\theta}}}{2(K+1)} \|\boldsymbol{\theta}\|_2^2}_{:= \mathcal{L}_{\boldsymbol{\theta}}} + \underbrace{\frac{\lambda_{\mathbf{B}}}{2MN} \|\mathbf{B}\|_F^2}_{:= \mathcal{L}_{\mathbf{B}}} + \underbrace{\frac{\lambda_{\boldsymbol{\tau}}}{2N} \|\mathbf{L}^{1/2} \boldsymbol{\tau}\|_2^2}_{:= \mathcal{L}_{\boldsymbol{\tau}}}. \quad (9)$$

The quantities  $\{\mathbf{x}_t^L(\boldsymbol{\Omega})\}_t$  and  $\{\hat{\mathbf{x}}_t\}_t$  in (9) correspond respectively to the outputs of DeepWave( $\boldsymbol{\Omega}, L$ ) and APGD( $\alpha, \lambda, \gamma$ ) with identical example input data  $\{(\hat{\Sigma}_t, \mathbf{x}_t^0)\}_t$ . The first term  $\frac{1}{T} \sum_{t=1}^T \mathcal{L}_t$  is a

<sup>15</sup>For a spherical array with diameter  $D = 30$  cm operating at 1 kHz,  $M \geq 90$  is sufficient.

data-fidelity term, which attempts to bring  $\hat{\mathbf{x}}_t$  and  $\mathbf{x}_t^L(\Omega)$  as close as possible from one another.<sup>16</sup> The additional terms  $\mathcal{L}_\theta, \mathcal{L}_\mathbf{B}, \mathcal{L}_\tau$  are smoothing *regularisers*, fighting against *overfitting*, a common issue in deep learning. Since the shrinkage operator  $\tau$  is defined over an irregular spherical tessellation, the smoothing term  $\mathcal{L}_\tau$  is defined via the Laplacian  $\mathbf{L} \in \mathbb{R}^{N \times N}$  associated to the connectivity graph of the tessellation, as is customary in graph signal processing (see appendix B.3).

Optimisation of (9) is carried out by *stochastic gradient descent (SGD)* with momentum acceleration [51]. Gradients of  $\mathcal{L}_t$  with respect to  $\theta, \mathbf{B}, \tau$  are efficiently evaluated using *reverse-mode algorithmic differentiation* [1, 25] and are given in algorithms 1 and 2 (see appendix F for a derivation). While random initialisation of neural-networks is a common practice in deep learning [51], this strategy failed for our specific architecture, leading to poor validation loss and considerably increased training times. Instead, we hence use the oracle parameters (7) to initialise SGD:

$$\theta^0 := \arg \min_{\theta \in \mathbb{R}^{K+1}} \|P_\theta(\mathbf{L}) - \mathcal{D}\|_F^2, \quad \mathbf{B}^0 := \sqrt{\frac{\alpha}{\beta}} \mathbf{A}, \quad \tau^0 := \frac{\lambda \alpha \gamma}{\beta} \mathbf{1}_N. \quad (10)$$

For greater numerical stability during training, we follow [41] and reparameterise the deblurring filter as  $P_\theta(\tilde{\mathbf{L}}) = \sum_{k=0}^K \theta_k T_k(\tilde{\mathbf{L}})$ , where  $T_k(\cdot)$  is the Chebychev polynomial of order  $k$  and  $\tilde{\mathbf{L}}$  is the normalised Laplacian with spectrum in  $[-1, 1]$  (see appendix B.3 for implementation details). Finally, we substitute the ReLU activation function by a scaled rectified tanh to avoid the exploding gradient problem [39].<sup>17</sup>

## 4 Experimental results

In this section, we compare the accuracy, resolution and runtime performance of DeepWave to DAS and APGD on real-world (RW) and simulated (SIM) datasets. More comprehensive dataset descriptions and additional results, including an ablation study, are provided in appendices G to I.

**Dataset 1 [36] (RW)** reproduces a conference room setup depicted in figs. 3a and 3b, where 8 people<sup>18</sup> are gathered around a table and speak either in turns or simultaneously (with at most 3 concurrent speakers). Recordings of the conversation are collected by the 48-element Pyramic array [46] (fig. 3f) positioned at the centre of the table. Since human speech is wide-band, the audible range [1500, 4500] Hz in the latter are pre-processed every 100 ms and split into 9 uniform bins to form a suitable training set  $\{(\tilde{\Sigma}_t, \hat{\mathbf{x}}_t, \mathbf{x}_t^0)\}_t$  of 2760 data points per frequency band for DeepWave (with  $N = 2234$ ). (See appendix G.2.) Frequency channels are processed independently by each algorithm. DeepWave is trained by splitting the data points into a training and validation set (respectively 80% and 20% in size). For each frequency band, we chose an architecture with 5 layers.

In fig. 3, figs. G.4 and H.3 respectively, we compare the accuracy and runtime of DeepWave, DAS and APGD. A video showing the evolution in time of DeepWave and DAS azimuthal sound fields (as in figs. 3a and 3b) is also available.<sup>19</sup> In terms of resolution, DeepWave and APGD perform similarly, outperforming DAS by approximately 27%. The mean contrast scores for DeepWave and DAS over the test set of Dataset 1 are 0.99 ( $\pm 0.0081$ ) and 0.89 ( $\pm 0.07$ ), respectively. Note that since the metrics used for assessing resolution and contrast<sup>20</sup> are not perfectly reflective of human-eye perception, the reported image quality improvements appear even more striking through visual inspection of the sound intensity fields (see for example fig. 3).

**Dataset 2 [43] (RW)** consists of 2700 template recordings from the Pyramic array taken in an anechoic chambre at an angular resolution of 2 degrees in azimuth and three different elevations (-15, 0, 15 degrees). Recordings contain both male and female speech samples to cover a wide audible range. The audio samples can be combined to simulate complex multi-source sound fields, hence we leverage this property to augment the dataset to 5700 distinct recordings with one, two, or three active speakers simultaneously. The raw time-series are then pre-processed as for Dataset 1 to obtain a

<sup>16</sup>in a mean relative squared-error sense.

<sup>17</sup>An alternative is to use a *truncated* ReLU. Given initialisation strategy 10, network training will still converge with similar step sizes as those used with tanh non-linearities.

<sup>18</sup>The 8 people are represented in the experiment by loudspeakers playing male and female speech samples.

<sup>19</sup>Available online: <https://www.youtube.com/watch?v=PwB3CS2rHdI>

<sup>20</sup>As is customary, resolution is measured as the *width at half-maximum* of the impulse response of the algorithms. Contrast is measured as the difference between the maximum and mean of the greyscale image.

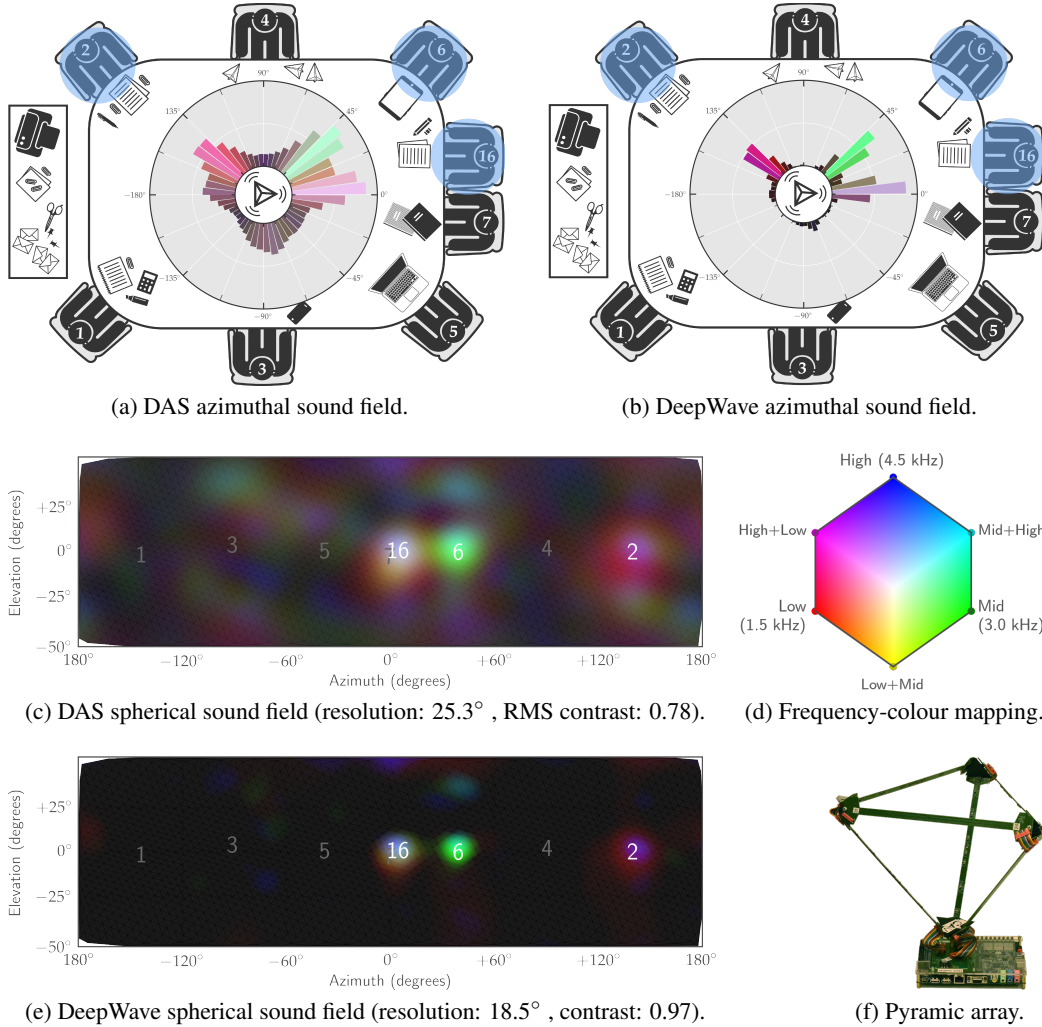


Figure 3: Snapshots at time  $t = 1.7$  s of the sound intensity fields produced by DeepWave and DAS for the Pyramic recordings with speakers 2, 6 and 16 active. Sound frequencies range from 1.5 to 4.5 kHz and were mapped to true colours (see fig. 3d, colour shades correspond to lower intensities). The spherical maps of DAS and DeepWave are plotted in figs. 3c and 3e, respectively. In figs. 3a and 3b we plot the azimuthal projections of figs. 3c and 3e, respectively.

training set of 151980 data points per frequency band (with  $N = 1568$ ). Network training is identical to that of Dataset 1, except that 10 azimuth directions are also withheld from the training set to assess how well the network generalises to emissions from unseen directions.

Figures 4a and 4b show sample DAS and DeepWave reconstructions with real sources from directions withheld from the training set. Similarly, fig. 4c shows sample reconstructions when the network is trained on real data but tested on synthetic narrow-band covariance matrices induced by sources from directions absent from the training set. In both cases we see that DeepWave outperforms DAS in resolution and contrast (i.e. sharper blobs and darker background).

**Dataset 3 (SIM)** finally is a dataset with recordings from a spherical microphone array using a narrow-band point-source data-model at 2 kHz [53]. The sources are randomly positioned over a  $120^\circ$  field-of-view, with up to 10 concurrent sources per recording. Experiment results available in fig. H.1 corroborate the real-data results, hence showing that DeepWave generalises well to a large number of sources with unconstrained positions. We further investigated in fig. H.2 the influence of network depth, and concluded that 5 or 6 layers are generally sufficient for the investigated dataset.



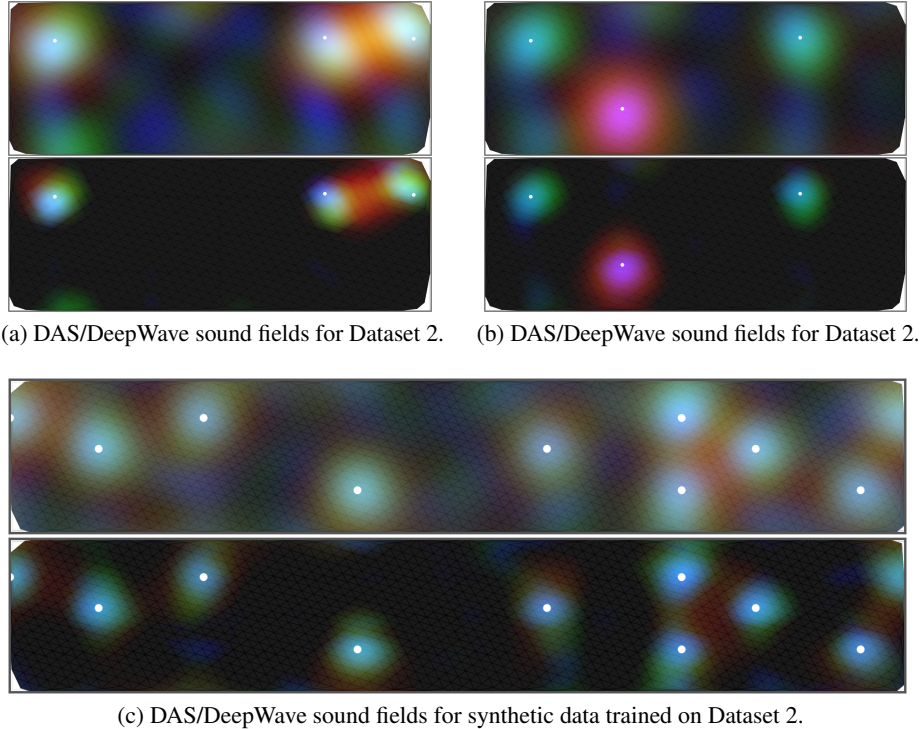


Figure 4: Snapshots of the sound intensity fields produced by DeepWave and DAS when trained on Dataset 2 (with 10 held-out source directions). Each subplot contains a DAS image (top) and a DeepWave image (bottom). The frequency color mapping is identical to fig. 3d. Figures 4a and 4b show azimuthal sound field slices on  $[-20^\circ, 150^\circ]$  using real-world covariance matrices with sources from unseen directions during training. Figure 4c shows a full  $360^\circ$  sound field on a synthetic covariance matrix from unseen directions during training. Elevations span  $[-15^\circ, +15^\circ]$ .

In terms of runtimes finally, DeepWave and DAS both reach real-time requirements (6.5 ms and 2.0 ms respectively), largely outperforming APGD (211 ms). (See fig. H.3 for more details.)

## 5 Conclusion

We introduced DeepWave, the first recurrent neural-network for real-time and high resolution acoustic imaging. It mimics iterative solvers from convex optimisation, while using the natural structure of acoustic imaging problems for efficient training and operation. Our real and simulated data experiments show DeepWave has similar computational speed to the state-of-the-art DAS imager with vastly superior resolution and contrast.

For future work, one of our goals is to make DeepWave *time-aware*, by training it on sequences of consecutive measurements in time. To this end, we plan to connect multiple DeepWave networks together, one for each time, and train them end-to-end. In such an architecture, the output neurons from one network would serve as initial neural state  $\mathbf{x}^0$  for the next network in line. This can be interpreted as warm-starting the network with the sound field estimated at the previous time instant. Additionally, we would like to propose a frequency-invariant DeepWave architecture, allowing to train a single network for all frequency bands. Properties of the oracle weights (7) suggest that this should be possible. This would considerably facilitate the training of the network, since the training set would be augmented and the number of trainable parameters reduced.

**Acknowledgments** We thank Erwan Zerhouni for useful discussions regarding network training and implementation details; and Ivan Dokmanić for insights on related works that inspired our approach. Finally we express our gratitude towards Robin Scheibler and Hanjie Pan for their openly-accessible real-world datasets [36, 43].

## References

- [1] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18:1–43, 2018.
- [2] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.
- [3] Jacob Benesty, Jingdong Chen, and Yiteng Huang. *Microphone array signal processing*, volume 1. Springer Science & Business Media, 2008.
- [4] Adrien Besson, Lucien Roquette, Dimitris Perdios, Matthieu Simeoni, Marcel Arditi, Paul Hurley, Yves Wiaux, and Jean-Philippe Thiran. A physical model of non-stationary blur in ultrasound imaging. *IEEE Transactions on Computational Imaging*, 2019.
- [5] Eric Bezzam, Robin Scheibler, Juan Azcarreta, Hanjie Pan, Matthieu Simeoni, René Beuchat, Paul Hurley, Basile Bruneau, Corentin Ferry, and Sepand Kashani. Hardware and software for reproducible research in audio array signal processing. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6591–6592. Ieee, 2017.
- [6] Jan Biemond, Reginald L Lagendijk, and Russell M Mersereau. Iterative methods for image deblurring. *Proceedings of the IEEE*, 78(5):856–883, 1990.
- [7] Thomas F Brooks and William M Humphreys. A deconvolution approach for the mapping of acoustic sources (damas) determined from phased microphone arrays. *Journal of Sound and Vibration*, 294(4-5):856–879, 2006.
- [8] Leon Brusniak, James Underbrink, and Robert Stoker. Acoustic imaging of aircraft noise sources using large aperture phased arrays. In *12th AIAA/CEAS Aeroacoustics Conference (27th AIAA Aeroacoustics Conference)*, page 2715, 2006.
- [9] Antonin Chambolle and Ch Dossal. On the convergence of the iterates of the “fast iterative shrinkage/thresholding algorithm”. *Journal of Optimization theory and Applications*, 166(3):968–982, 2015.
- [10] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1256–1272, 2017.
- [11] Ning Chu, José Picheral, Ali Mohammad-Djafari, and Nicolas Gac. A robust super-resolution approach with sparsity constraint in acoustic imaging. *Applied Acoustics*, 76:197–208, 2014.
- [12] Zhigang Chu and Yang Yang. Comparison of deconvolution methods for the visualization of acoustic sources based on cross-spectral imaging function beamforming. *Mechanical Systems and Signal Processing*, 48(1-2):404–422, 2014.
- [13] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, pages 3844–3852, 2016.
- [14] Simon Foucart and Holger Rauhut. A mathematical introduction to compressive sensing. *Bull. Am. Math.*, 54:151–165, 2017.
- [15] Krzysztof M Gorski, Eric Hivon, Anthony J Banday, Benjamin D Wandelt, Frode K Hansen, Mstvos Reinecke, and Matthia Bartelmann. Healpix: a framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759, 2005.

- [16] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 399–406. Omnipress, 2010.
- [17] Harshit Gupta, Kyong Hwan Jin, Ha Q Nguyen, Michael T McCann, and Michael Unser. Cnn-based projected gradient descent for consistent ct image reconstruction. *IEEE transactions on medical imaging*, 37(6):1440–1453, 2018.
- [18] RK Hansen and PA Andersen. A 3d underwater acoustic camera—properties and applications. In *Acoustical Imaging*, pages 607–611. Springer, 1996.
- [19] Doug P Hardin, Timothy Michaels, and Edward B Saff. A comparison of popular point configurations on  $s^2$ . *Dolomites Research Notes on Approximation*, 9(1), 2016.
- [20] Paul Hurley and Matthieu Simeoni. Flexibeam: analytic spatial filtering by beamforming. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2877–2880. Ieee, 2016.
- [21] Paul Hurley and Matthieu Simeoni. Flexarray: Random phased array layouts for analytical spatial filtering. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3380–3384. Ieee, 2017.
- [22] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.
- [23] KG Jinadasa. Applications of the matrix operators vech and vec. *Linear Algebra and its Applications*, 101:73–79, 1988.
- [24] Charles H Jones and George A Gilmour. Acoustic camera apparatus, August 5 1975. US Patent 3,898,608.
- [25] Sepand Kashani. Optimization notes. page 6, 2019.
- [26] K Kim, N Neretti, and N Intrator. Mosaicing of acoustic camera images. *IEE Proceedings-Radar, Sonar and Navigation*, 152(4):263–270, 2005.
- [27] Hamid Krim and Mats Viberg. Two decades of array signal processing research. *IEEE signal processing magazine*, 1996.
- [28] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
- [29] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [30] Huan Li, Yibo Yang, Dongmin Chen, and Zhouchen Lin. Optimization algorithm inspired deep neural network structure design. *arXiv preprint arXiv:1810.01638*, 2018.
- [31] Jingwei Liang and Carola-Bibiane Schönlieb. Faster fista. *arXiv preprint arXiv:1807.04005*, 2018.
- [32] Michael Lustig, David Donoho, and John M Pauly. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.
- [33] Michael T McCann, Kyong Hwan Jin, and Michael Unser. Convolutional neural networks for inverse problems in imaging: A review. *IEEE Signal Processing Magazine*, 34(6):85–95, 2017.
- [34] Andy Meyer and Dirk Döbler. Noise source localization within a car interior using 3d-microphone arrays. *Proceedings of the BeBeC*, pages 1–7, 2006.
- [35] Volker Michel. Lectures on constructive approximation. *AMC*, 10:12, 2013.

- [36] Hanjie Pan, Robin Scheibler, Eric Bezzam, Ivan Dokmanić, and Martin Vetterli. Audio speech recordings used in the paper FRIDA: FRI-based DOA Estimation for Arbitrary Array Layout, March 2017. This work was supported by the Swiss National Science Foundation grant 20FP-1 151073, LABEX WIFI under references ANR-10-LABX-24 and ANR-10-IDEX-0001-02 PSL\* and by Agence Nationale de la Recherche under reference ANR-13-JS09-0001-01.
- [37] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239, 2014.
- [38] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *IEEE signal processing magazine*, 20(3):21–36, 2003.
- [39] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318, 2013.
- [40] Dimitris Perdios, Adrien Besson, Marcel Arditi, and Jean-Philippe Thiran. A deep learning approach to ultrasound image recovery. In *2017 IEEE International Ultrasonics Symposium (IUS)*, pages 1–4. Ieee, 2017.
- [41] Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. Deepsphere: Efficient spherical convolutional neural network with healpix sampling for cosmological applications. *Astronomy and Computing*, 2019.
- [42] Boaz Rafaely. *Fundamentals of spherical array processing*, volume 8. Springer, 2015.
- [43] Scheibler Robin. Pyramic Dataset : 48-Channel Anechoic Audio Recordings of 3D Sources, March 2018. The author would like to acknowledge Juan Azcarreta Ortiz, Corentin Ferry, and René Beuchat for their help in the design and usage of the Pyramic array. Hanjie Pan, Miranda Kreković, Mihailo Kolundzija, and Dalia El Badawy for lending a hand, or even two, during experiments. Finally, Juan Azcarreta Ortiz, Eric Bezzam, Hanjie Pan and Ivan Dokmanić for feedback on the documentation and dataset organization.
- [44] Justin Romberg. Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):14–20, 2008.
- [45] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [46] Robin Scheibler, Juan Azcarreta, René Beuchat, and Corentin Ferry. Pyramic: Full stack open microphone array architecture and dataset. In *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 226–230. IEEE, 2018.
- [47] David Shuman, Sunil Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 3(30):83–98, 2013.
- [48] Pieter Sijtsma. Clean based on spatial source coherence. *International journal of aeroacoustics*, 6(4):357–374, 2007.
- [49] Matthieu Martin Jean-Andre Simeoni and Paul Hurley. Graph spectral clustering of convolution artefacts in radio interferometric images. Technical report, 2019.
- [50] Jian Sun, Huibin Li, Zongben Xu, et al. Deep admm-net for compressive sensing mri. In *Advances in neural information processing systems*, pages 10–18, 2016.
- [51] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.
- [52] Michael Unser, Julien Fageot, and Harshit Gupta. Representer theorems for sparsity-promoting l1 regularization. *IEEE Transactions on Information Theory*, 62(9):5167–5180, 2016.
- [53] Alle-Jan van der Veen and Stefan J Wijnholds. Signal processing tools for radio astronomy. In *Handbook of Signal Processing Systems*, pages 421–463. Springer, 2013.

- [54] Yves Wiaux, Laurent Jacques, Gilles Puy, Anna MM Scaife, and Pierre Vanderghenst. Compressed sensing imaging techniques for radio interferometry. *Monthly Notices of the Royal Astronomical Society*, 395(3):1733–1742, 2009.
- [55] Stefan J Wijnholds and Alle-Jan van der Veen. Fundamental imaging limits of radio telescope arrays. *IEEE Journal of Selected Topics in Signal Processing*, 2(5):613–623, 2008.
- [56] Li Xu, Jimmy SJ Ren, Ce Liu, and Jiaya Jia. Deep convolutional neural network for image deconvolution. In *Advances in Neural Information Processing Systems*, pages 1790–1798, 2014.
- [57] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.