SINR: Deconvolving Circular SAS Images Using Implicit Neural Representations

Albert Reed, Thomas Blanford, Daniel C. Brown, Senior Member, IEEE, and Suren Jayasuriya, Member, IEEE

ABSTRACT—Circular synthetic aperture sonars (CSAS) capture multiple observations of a scene to reconstruct high-resolution images. We can characterize resolution by modeling CSAS imaging as the convolution between a scene's underlying point scattering distribution and a system-dependent point spread function (PSF). The PSF is a function of the system bandwidth and determines a fixed degree of blurring on reconstructed imagery. In theory, deconvolution overcomes bandwidth limitations by reversing the PSF-induced blur and recovering the scene's scattering distribution. However, deconvolution is an ill-posed inverse problem and sensitive to noise. We propose an optimization method that leverages an implicit neural representation (INR) to deconvolve CSAS images. We highlight the performance of our SAS INR pipeline, which we call SINR, by implementing and comparing to existing deconvolution methods. Additionally, prior SAS deconvolution methods assume a spatially-invariant PSF, which we demonstrate yields subpar performance in practice. We provide theory and methods to account for a spatially-varying CSAS PSF, and demonstrate that doing so enables SINR to achieve superior deconvolution performance on simulated and real acoustic SAS data.

INDEX TERMS—Deconvolution, inverse problems, neural networks, optimization, synthetic aperture sonar.

I. INTRODUCTION

SYNTHETIC aperture sonar (SAS) is an important technology for capturing high-resolution images of the seafloor [1], [2]. SAS enhances long-track image resolution by coherently combining acoustic measurements taken from a platform moving along a known track [3], [4]. A variety of algorithms exist for forming images from SAS acoustic measurements [5], [6], [7], [8]. These reconstruction algorithms aim to produce high quality imagery to support underwater visualization tasks including target localization [9], monitoring man-made infrastructure [10], and studying underwater ecologies [11].

In stripmap SAS imaging, the along-track resolution is proportional to transducer size and aperture length, while the cross-track resolution is proportional to transmit waveform bandwidth [3], [4]. In this case, cross-track and along-track resolutions have simple analytical expressions [4]. However, deriving the imaging resolution is usually system dependent as many platforms deviate from a linear path or use custom transducer arrays.

A general method for describing SAS image resolution is its imaging point spread function (PSF). The SAS PSF is the reconstructed image of a point scatterer within the scene. If the PSF is shift-invariant, the point scatterer can be centered in the scene for simplicity. A spatially invariant assumption and representing the scene as a distribution of point scatterers allows SAS imaging to be modeled as the convolution of the PSF with the scene [12], [13], [14], [15]. The shape of the PSF induces a system-dependent blurring on the scene and describes the imaging resolution.

Practical barriers exist for deploying SAS systems capable of resolving arbitrarily fine spatial frequencies in the imaging scene. These limitations are born from physical constraints in the electrical and mechanical hardware used to construct the sensor. In particular, SAS arrays are band-limited by the electromechanical properties of the sonar transducers and the maximum data rate of the digitizing electronics [2], [17]. In theory, we can address these limitations in post-processing by deconvolving the PSF from computed SAS images [13], [15], [18].

Deconvolution is a classic problem in many sensing fields including optical imaging [19], [20], [21], [22], [23] and seismology [24], [25]. Fewer works explore deconvolution for SAS [13], [15], [26], and we did not find any works in literature that benchmark competing deconvolution methods on SAS datasets. Further, the application of deep neural networks, which often yield state-of-the-art results for inverse imaging problems [27], remains unexplored for performing SAS deconvolution. Our work serves to fill this gap in existing literature.

In this paper, we investigate SAS deconvolution for circular SAS (CSAS) collection geometries. CSAS enables improved resolution and reduced speckle noise by maximizing the number of angular look directions at a target [2], [28], [29], [30], [31], [32].

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
[31]. We develop a novel pipeline that uses an implicit neural representation (INR) to deconvolve SAS images. Our SAS INR pipeline, which we call SINR, uses knowledge of the CSAS PSF to optimize a network with an analysis-by-synthesis cost function. In particular, we use an imaging model combined with a cost function to fit the network parameters, i.e., its weights. We emphasize that the proposed method optimizes a new network per image — networks are not being trained for later deployment on unseen data.

We identify that existing SAS deconvolution methods do not account for the spatially-varying nature of the SAS PSF, which limits their application. To address this, we design a neural network capable of predicting a complex scattering field to perform a coherent deconvolution that accounts for a scene’s spatially-varying phase. We evaluate our pipeline on simulated measurements created with a point scattering model and real measurements captured with an in-air SAS system called AirSAS [16] (see Fig. 1). Our specific contributions are as follows:

1) We propose SINR, a pipeline that optimizes implicit neural networks to coherently deconvolve CSAS images.

2) We present theory and analysis showing that SINR accounts for the CSAS PSF’s spatially-varying phase in its reconstruction.

3) We implement prior deconvolution approaches to serve as competing methods for SINR.

4) We build and release synthetic and real SAS datasets and code implementations in Python for benchmarking purposes.

We characterize SINR and competing deconvolution methods on simulated and real acoustic data by varying noise and bandwidth conditions in our experiments. Additionally, we provide code for our open-source simulator, SINR implementation, and competing methods for benchmarking and reproducibility of research.

This work is an extension of an earlier conference paper [32]. We have extended the paper’s results by implementing new methods, theory, and analysis to perform CSAS deconvolution, including handling the spatially-varying phase of the CSAS PSF. Additionally, we systematically benchmarked our proposed approach against implemented competing methods on simulated and real datasets.

II. BACKGROUND AND RELATED WORK

This section discusses conventional SAS image reconstruction methods and common post-processing techniques. Prior work on general image and sonar/SAS-specific deconvolution techniques are presented. Finally, we provide background on the implicit neural representations networks that SINR leverages to perform CSAS deconvolution.

A. SAS Image Reconstruction

SAS imaging algorithms typically process SAS data in either the time or frequency domain by mapping backscattered measurements to pixels in the scene [3], [5], [6], [7], [8]. Several custom reconstruction algorithms exist for CSAS [7], [33], [34], many of which leverage algorithms from medical imaging due to the geometry’s similarities to computed tomography [28]. In this work, we use the time-domain method titled delay-and-sum (DAS), which uses time-of-flight measurements between transducer and scene pixels to backproject measurements onto a scene. We choose this method because it is capable of image reconstruction for near arbitrary SAS aperture geometries and works for both near-field and far-field reconstructions.

Post-reconstruction algorithms to correct uncertainties and artifacts in SAS images is an active area of research. The majority of these algorithms estimate the platform track and motion to correct imaging errors [4], [34], [35], [36], [37], [38], [39], [40] and account for environment noise [4], [5], [41], [42]. While our method addresses a different problem, we consider it to be a post-reconstruction algorithm because it requires a DAS reconstructed image as input.

B. Deconvolution

Many works tackle blind (unknown PSF) [43] and non-blind (known PSF) deconvolution for camera-based imaging [21], microscopy [22], astronomy [23], and seismology [24]. Deconvolution is ill-posed since a solution might not exist, might not be unique, or might be unstable when small amounts of noise exist in the data [44]. As such, deconvolution methods leverage priors such as total variation (TV) and gradient regularization to recover favorable solutions [44], [45], [46]. We implement a deconvolution method based on BREGMAN ADMM and gradient regularization to better contextualize the performance of our method within this body of work.

de Heering et al. [15] presented the first formulation of the deconvolution problem for SAS and propose a deconvolution algorithm called BREMEN. The BREMEN algorithm assumes a spatially-invariant SAS PSF that does not consider the phase of the underlying signals. To highlight the limitations of this assumption, we implement BREMEN to serve as a competing method for the proposed approach. In subsequent work, Marston et al. [26] presents a Wiener deconvolution algorithm which preserves phase information and enhances image features. As
such, we implement a Wiener deconvolution method to serve as an additional competing method.

Pailhas et al. [13] present a sampling scheme for ensuring the SAS PSF is spatially-invariant in circular SAS and introduce a deconvolution method based on atom wavelets. However, the proposed method requires sampling a new PSF for each image pixel in the scene which is not computationally tractable for practical scenes containing on the order of 100,000 pixels. In later work, the same authors [14] derive analytical expressions for the CSAS PSF and describe theory that we leverage in our method’s formulation.

Other related sonar/acoustic deconvolution works include correcting image aberrations caused by a SAS platform exceeding its maximum coverage rate [47], [48], the Richardson-Lucy algorithm for deconvolving real-aperture sonar images [49], and DAMAS [50], [51] which uses an iterative non-negative least-squares solver to deconvolve measurements captured with phased microphone arrays. In contrast to all these methods, we utilize implicit neural representations to perform coherent SAS deconvolution.

A recent result in the computer vision literature demonstrates that untrained, deep convolutional neural networks (CNNs) are useful in optimization settings where an objective function can be formulated using physics-based constraints. In contrast with the traditional machine learning framework, where a model is optimized on a training dataset and evaluated on a test dataset, many works show that CNNs are useful in inverse optimization problems where the goal is to recover an image from measurements. In Ulyanov et al. [52], the authors develop a framework, called the deep image prior (DIP) for leveraging untrained networks in solving ill-posed, inverse imaging problems, including deconvolution. Deep image priors have been successfully applied to reconstructing positron emission tomography [53], computed tomography [54], and compressively sensed images [55], and partially inspired our application of neural networks to SAS deconvolution. We implement a DIP network to serve as a competing method for our proposed approach.

C. Implicit Neural Representations

Our approach to SAS deconvolution leverages a new type of neural network called an implicit neural representation (INR) [56]. INRs are typically constructed from fully-connected neuron layers and are tasked with learning a function that maps input coordinates to physical properties in the scene (e.g., the acoustic reflectivity at \((x, y, z)\)). In this sense, networks are optimized to encode an implicit representation of scenes within their weights. These networks have shown much promise at modeling space and time fields, as discussed in this comprehensive survey by Xie et al. [57]. In particular, these models have achieved state-of-the-art results on inverse imaging problems, which inspired our application of them to SAS deconvolution.

In the fully-connected layer configuration, INRs use a positional encoding transformation that maps input coordinates to a high dimensional space using a Fourier basis. This transformation is vital for enabling INRs to overcome spectral bias and fit high frequency information [58]. This positional encoding can be tuned to regularize the spatial frequencies the INR reconstructs and bias the reconstruction. In our experiments, we show that turning the an INR’s positional encoding is key to achieving high quality CSAS deconvolution results. As such, this is the architecture we recommend for the proposed deconvolution method.

In literature, the success of INRs (and other untrained network architectures) is attributed to their inductive bias [52], [58]. Inductive bias is a term used to describe the observation that over-parameterized networks have an affinity for finding smooth solutions. In inverse imaging, this manifests as networks having a natural affinity (i.e., a bias) for estimating images with visually salient features while maintaining a high impedance to noise. We note that other deconvolution methods create a bias towards smooth solutions adding regularization, as discussed in Section II-B. Theoretical justification for networks’ inductive bias is an active research area [59], [60]. While a complete mathematical analysis for the inductive bias of neural networks (and particularly how the network structure influences the bias) is outside the scope of this paper, we use the term to describe the performance of our INR relative to competing methods throughout this manuscript.

We note that other INR architectures that do not use a positional encoding have been proposed. Notably, SIREN is an architecture that uses special layers with periodic activation functions to enable the network to fit high frequencies [61]. We also implement our method using this architecture, but find the results on real data are suboptimal since the method lacks a tunable positional encoding parameter.

III. PROPOSED APPROACH

We begin this section by describing our imaging model for sensing and reconstructing CSAS data. We then represent this model as a convolution and present our deconvolution approach. We refer the reader to Fig. 2 for an illustration of our reconstruction pipeline. We define commonly used notation in Table I.

A. Point Scattering Model and DAS Reconstruction

We consider a CSAS geometry where a ring of transducers circles a scene \((x, y, z)\) of omnidirectional point scatterers with
scalar response coefficients $\sigma(x, y, z)$ where $\sigma : \mathbb{R}^3 \mapsto \mathbb{R}^+$. To aid in computational tractability, we model our scene of interest and PSF in two dimensions (i.e., $(x, y)$ by assigning $z = z_0$ (it remains a future direction of research to consider the full 3D SAS deconvolution problem and anisotropic scatterers). As such, we define radial and azimuthal Fourier coordinates as $\rho$ and $\phi$, respectively.

We denote the transducer positions as $T_{\theta_i} = \{(x_{\theta_i}, y_{\theta_i}, z_{\theta_i}) \in \mathbb{R}^3 : \theta_i \in [0, 2\pi) \text{ radians}\}$. To avoid aliasing, we sample in incremental angular steps $\Delta \theta \leq \frac{2\pi}{r'}$ radians where $\theta_{\min}$ is the transmit waveform’s minimum wavelength and $r'$ is the circle radius encapsulating the scene of interest [16]. Each transducer transmits a linear frequency modulated (LFM) chirp with waveform parameters denoted as $w(t) = B(t) \cos(2\pi \frac{\Delta f}{2} t^2 + 2\pi f_{\text{start}} t)$ where the bandwidth $\Delta f = |f_{\text{start}} - f_{\text{stop}}|$ is the range between the start and stop frequencies, $T$ is the pulse duration, $t$ is time, and $B(t)$ is a windowing function. We use a Tukey window with a cosine fraction of 0.1 for our experiments. Additionally, we define the center frequency as $f_c = \frac{f_{\text{start}} + f_{\text{stop}}}{2}$.

Using these definitions, we define a point scattering model

$$S_{T_{\theta_i}}(t) = \iiint_{x,y} \sigma(x, y, z_0) \, w \left( t - 2\frac{||T_{\theta_i} - (x, y, z_0)||_2}{c} \right) \, dx \, dy + n(t), \quad (1)$$

where $S_{T_{\theta_i}} : \mathbb{R} \mapsto \mathbb{R}$ are the amplitude and phase modulated transmit waveform measurements, $c$ is the speed of sound, and $n(t) \sim \mathcal{N}(0, \eta^2)$ is i.i.d. Gaussian noise.\footnote{For a 3D scene, this equation would be a triple integral with $z$ as a free variable as well.}

We coherently process measurements by match-filtering with the transmitted waveform $S_{T_{\theta_i}}^{CC}(t) = R\{F^{-1}(S_{T_{\theta_i}} \hat{w}^*)\}$ and computing the complex-valued analytic signal $S_{T_{\theta_i}}^A(t) = S_{T_{\theta_i}}^{CC}(t) + j\mathcal{H}(S_{T_{\theta_i}}^{CC}(t))$ [5], [62]. We reconstruct images using delay-and-sum (DAS) reconstruction given by

$$I(x, y, z_0) = \int_{T_{\theta_i}} S_{T_{\theta_i}}^A \left( \frac{2||T_{\theta_i} - (x, y, z_0)||_2}{c} \right) \, dT_{\theta_i}, \quad (2)$$

where $I : \mathbb{R}^3 \mapsto \mathbb{C}$ is the complex scattering amplitude at pixel $(x, y, z_0)$, enabling easy computation of the image amplitude $|I|$ and phase $\angle I$.

**B. Forward Imaging Model**

The DAS image $I$ reconstructs a blurred complex image of the scattering distribution $\sigma$. The blurring is a function of the PSF shape. We refer to Fig. 3 to illustrate examples of the CSAS PSF. For fractional bandwidth ($\frac{\Delta f}{f_c}$) less than 1, the behavior of the PSF near its peak is dominated by the center frequency ($f_c$) oscillations of a Bessel function [13]. The width of the oscillation nearest the peak are approximately equal in the two PSFs because they have the same center frequency. Away from the peak of the PSF, however, the bandwidth and window in the waveform dominate the behavior. Increasing the bandwidth of the waveform causes the oscillations of the PSF to decay in amplitude more rapidly. This can be observed in the relative structure of the PSF’s away from the peak for 20 kHz than 5 kHz.

We wish to correct PSF-induced blur by solving an inverse problem that recovers the original scattering distribution from the DAS image. As noted in [13], [15], [18], if we assume the PSF is spatially-invariant, we can model the DAS image as the...
convolution of scene scatterers $\sigma$ with the CSAS PSF:

$$I = \sigma * I_{\text{PSF}},$$

where $I_{\text{PSF}}$ is the DAS reconstruction of simulated backscattered measurements from a point centered in the scene.

While assuming a spatially invariant PSF drastically simplifies deconvolution [63], [64], we find that doing so can yield subpar performance in practice, particularly if the goal is to perform coherent deconvolution. Fig. 4 displays the phase $\angle I_{\text{PSF}}$ of a simulated PSF at three locations in an AirSAS scene (see Section V for AirSAS scene geometry details). We observe that the centered PSF is dominantly real, while the offcenter PSF (either translated on the imaging plane or above the plane such that $z > z_0$) contains significant imaginary components. Given these observations, we anticipate that existing SAS deconvolution methods like BREMEN [15] that assume a spatially-invariant PSF will fail to properly deconvolve scatterers offset from the scene center, as demonstrated in future experiments.

Observations from Fig. 4 are supported by an analytical model of the CSAS PSF created by Pailhas et al. that gives the spectrum of a centered PSF as [13]

$$\hat{I}_{\text{PSF, center}}(\rho) \approx \pi^2 \sqrt{\frac{2 \alpha_0 \sigma}{k}} \exp \left[ -2a_0^2 (k - \pi \rho)^2 \right],$$

where $\rho$ is the radial polar coordinate, $k = \frac{2\pi}{\lambda}$ where $\lambda$ is wavelength, and $\alpha_0$ and $\sigma$ are functions of the waveform bandwidth and duration (the interested reader may refer to [14] for their exact form). Our numerical simulation visualizes this expression, as it shows the centered PSF’s spectrum is conjugate symmetric and thus strictly real in the spatial domain.

In follow-up work, Pailhas et al. give an equation for the general PSF (center and off-center) as [14]:

$$\hat{I}_{\text{PSF}}(\rho, \phi) = \sqrt{2} \sigma \alpha_0 \exp \left[ -2a_0^2 (\pi \rho - k)^2 \right] \exp \left[ j \phi \right] \exp \left( j \phi \right) \left( \phi + \theta_0 + \frac{\pi}{2} \right).$$

This equation quantifies our simulation observations by describing that the off-center PSF differs from its centered counterpart by multiplication with a phase $\phi' = 2\pi \rho R \cos(\theta_0 - \phi)$ and angular weighting term $g(\phi + \theta_0 + \frac{\pi}{2})$ that scale with the PSFs position (given by $(R, \theta_0)$ in polar coordinates) relative to the scene center.

### C. Deconvolution With INRs

Naively deconvolving CSAS images using the centered PSF yields the inverse filter,

$$\sigma = I *^{-1} I_{\text{PSF}} = \mathcal{F}^{-1} \left( \frac{\hat{I}}{\hat{I}_{\text{PSF}}} \right),$$

where $*^{-1}$ denotes a deconvolution operator, typically implemented in the frequency domain by dividing the image spectrum by the PSF spectrum. However, this division amplifies high frequency noise, and thus more robust deconvolution algorithms are needed. Additionally, given the discussion above, the CSAS PSF has a spatially varying phase that should be accounted for to yield an accurate deconvolution.

The brute-force method to account for a spatially-varying PSF is to recompute the PSF at each pixel within the scene, as done in [13]. However, this approach becomes computationally expensive for large scenes. Thus, given the offcenter PSF is equivalent to the centered PSF multiplied with an exponential phase term $\exp[j\phi']$ and angular weighting term $g(\cdot)$, we propose tasking a neural network with predicting a complex-valued scene that is convolved with the real, centered PSF. This approach utilizes a single PSF to perform the convolution, and gives the responsibility of handling its spatially varying properties to our neural network predicting a complex scene of scatterers. Specifically, we approximate the general PSF equation in the frequency domain as

$$\hat{I}_{\text{PSF}} \approx \hat{I}_{\text{PSF, center}} \exp \left[ j \phi' \right] \exp \left( j \phi + \theta_0 + \frac{\pi}{2} \right),$$

$$\approx \hat{I}_{\text{PSF, center}} \cdot \gamma.$$  \hspace{1cm} (7)

Then, our convolution in the Fourier domain becomes the centered PSF multiplied with the weighting terms and the scene

$$\hat{I} = \sigma \cdot \hat{I}_{\text{PSF}} = \sigma \cdot \hat{I}_{\text{PSF, center}} \cdot \gamma,$$

which in the spatial domain gives

$$I = \sigma * \gamma \cdot \hat{I}_{\text{PSF, center}} = \sigma \cdot \hat{I}_{\text{PSF, center}}.$$  \hspace{1cm} (9)

In other words, rather than recompute the PSF at each spatial location, we propose predicting a complex valued scene of scatterers $\hat{\sigma} \in \mathbb{C}$, such that the convolution of the centered PSF with the predicted scatterers results in a complex-valued image. The scattering magnitude $|\hat{\sigma}|$ yields our deconvolution result. The advantage of our approach is that it harmonizes with fast convolution implementations since the PSF effectively remains spatially invariant in our model. However, the network is now tasked with the harder task of predicting a complex field $\hat{\sigma}$ that
is the bandwidth (10) are fed to the network such that the output INR and \(\cos(Bz)\) of the network [52].

encapsulates both the PSF’s spatially varying properties and the scene’s scattering distribution. As a result, the predicted image \(\hat{\sigma}\) may not theoretically match the ground truth scattering distribution \(\sigma\). Nevertheless, we find that this strategy facilitates both reconstruction accuracy and computational simplicity in practice.

We propose optimizing the weights of an implicit neural representation (INR) to invert (9). Specifically, we solve an optimization of the form

\[
\min_p \| |\text{INR}_p(z) * I_{PSF_{center}} - I||_2, \tag{10}
\]

where \(z = (x, y) \in \mathbb{R}^{N \times N}\) are the discretized coordinates of the scene and \(\text{INR}_p\) is the INR network parameterized with weights \(p\). As seen in other works [52], [56], we normalize input coordinates \((x, y)\) within the range \([-1, 1]\) to ensure the magnitude of the network weights are independent of the scene’s scale.

Equation 10 defines an analysis-by-synthesis optimization [65], [66] that tasks an INR with estimating the complex-valued scattering distribution \(\hat{\sigma}\) at every spatial location \((x, y)\) that, when convolved with the centered PSF, best fits the DAS image.² Key to our approach, we optimize the weights of a network, rather than scene values directly, such that the network outputs a scene that minimizes the reconstruction error. The strategy leverages the inductive bias of the network [52]. The network is biased towards estimating functions that exhibit minimal noise artifacts and conform to image priors such as edges and shapes.

We construct our INR network using 7-8 fully-connected layers with ReLU [67] non-linearities. We find that the deconvolution result changes slightly depending on the number of layers. In particular, we find that 7 layers works best for the real results and 8 layers works best for the simulated results. We follow the strategy described in Mildenhall et al. [58] to transform input coordinates \(z\) with random Fourier features given by \(\omega(z) = [\cos(2\pi \kappa Bz), \sin(2\pi \kappa Bz)]\), where \(\cos\) and \(\sin\) are performed element-wise; \(B\) is a matrix randomly sampled from a Gaussian distribution \(\mathcal{N}(0, I)\), and \(\kappa\) is the bandwidth factor. For 2D coordinates, the matrix \(B\) has shape \(2 \times L\). As such, each \((x, y)\) coordinate is projected to a higher dimension \(L\). For all experiments, we set \(L = 256\), a value that is commonly used in the literature [56], [58]. The \(\kappa\) parameter, often referred to as the INR’s bandwidth parameter in literature, influences the INR’s ability to model high frequencies and suppress noise [58]. In practice, we choose \(\kappa\) such that it accurately reconstructs features of the target while suppressing side-lobe energy and noise. As such, the parameter can be both scene and sensor dependent, and it is necessary to perform a parameter sweep to obtain the optimal reconstruction. The transformed coordinates \(\omega(z)\) are fed to the network such that the output \(\text{INR}_p(\omega(z))\) represents the scattering field, and we optimize the network weights by optimizing (10) using backpropagation. Optimizing (10) on a 400 × 400 CSAS scene takes under 10 minutes on an A100 graphical processing unit (GPU) [68].

**D. Competing Methods for Comparison**

We implement several deconvolution methods to serve as competing algorithms to contextualize the performance of SINR. For all methods, we identify learning and regularization weights for the network and gradient descent methods, and optimal SNR parameters for the Wiener filter, that maximize quantitative performance on the simulated data and qualitative performance on real data. We run all iterative methods until convergence and report the images that achieve the best PSNR or qualitative performance. We note that such a comparative analysis for SAS deconvolution on both simulated and real data has not been conducted before in the literature.

---

²We note that our deconvolution method operates on the DAS image rather than the waveforms. It remains a direction of future work to investigate deconvolving the waveforms.
1) **Wiener Deconvolution**: Our first competing method is Wiener Filter deconvolution, also proposed by Marston et al. [26] for SAS deconvolution, given by

\[
\sigma = \mathcal{F}^{-1} \left( \frac{I_{\text{PSF,cone}}^* I}{|I_{\text{PSF,cone}}^*|^2 + \alpha} \right),
\]

where \( \alpha \) is the mean power spectral density of the noise [69]. We note that Wiener deconvolution can run in seconds on a modern CPU and is thus our fastest implemented deconvolution method.

2) **Gradient Descent Deconvolution**: SINR leverages the inductive bias of an INR by backpropagating an analysis-by-synthesis loss to the network weights, rather than the scene scatterers. To test convergence without this inductive bias, we implement a method that optimizes the scene scatterers directly using a gradient descent optimization,

\[
\min_{\hat{\sigma}} \| \hat{\sigma} * I_{\text{PSF,cone}} - I \|_2.
\]

Additionally, we use this optimization to serve as a platform for implementing popular smoothness priors total variation (TV) and gradient regularization [45], [46],

\[
\min_{\hat{\sigma}} \| \hat{\sigma} * I_{\text{PSF,cone}} - I \|_2 + \beta R(\hat{\sigma}),
\]

where \( R(\cdot) \) is either the total variation or gradient regularization smoothness operator and \( \beta \) is the regularization parameter. In our experiments, we term these two methods as GD + TV and GD + Grad Reg., respectively. For all gradient descent experiments, we experiment with initializing point scatterers to the uniform distribution and to a Cartesian grid of coordinates (similar to the INR initialization). We find the latter yields marginally superior quantitative results and use it for all experiments. We optimize the gradient descent methods using the stochastic gradient descent (SGD) optimization implemented in PyTorch and obtain optimal performance with a learning rate of 10e-1 and momentum factor of 0.9. Our implemented gradient descent methods have similar runtimes to SINR, taking approximately 10 minutes to converge on a 400 \times 400 CSAS scene using an A100 GPU.

3) **BREMEN Deconvolution**: We implement the BREMEN [15] algorithm to serve as an additional competing method. BREMEN solves for the scene using the method of successive approximations [70]. As discussed in previous text, this method ignores phase information by assuming the PSF is spatially invariant. Our BREMEN implementation method takes under 5 minutes to converge on a modern CPU.

4) **DIP Deconvolution**: Our final competing method is a neural network approach based on the deep image prior (DIP) [52]. The DIP shows competitive results on many ill-posed inverse imaging problems as discussed in Section II-B. We implement a bottle-neck, U-net convolutional neural network architecture with skip-connections as presented in [52]. The DIP network ingests a \( M \) dimensional vector \( \mathbf{Z} \in \mathbb{R}^M \) taken from the uniform distribution \( \mathbf{Z} \sim \mathcal{U}(0, 1) \). Identical to SINR, we optimize the network weights using backpropagation to minimize (10). Our DIP network implementation takes approximately 20 minutes to converge on a 400 \times 400 CSAS scene using an A100 GPU.

IV. **Simulation Results**

Our proposed and competing methods are tasked with recovering the original scattering distribution from each simulated DAS image. We use three metrics to measure the difference between ground truth and deconvolved images. Namely, we consider the peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) [71], and Learned perceptual image patch (LPIPS) metric [72]. The PSNR and SSIM metrics capture pixel-level differences and the LPIPs metric quantifies perceptual similarity [73].

A. **Deconvolution Results and Noise Sweep**

Our first experiment is shown in Fig. 5 where we characterize the performance of deconvolution methods under varying degrees of measurement noise.\(^3\) To perform this experiment, we sweep the measurement noise between no noise and 17 dB PSNR while keeping transmit waveform bandwidth fixed to 20 kHz\(^4\). Specifically, we normalize the intensity of all SASSED images to the range [0, 1] and define the waveform PSNR as

\[
\text{PSNR} = 10 \log_{10} \left( \frac{1}{\sqrt{\eta}} \right)^2,
\]

where \( \eta \) is the variance of the measurement noise in (1). In Fig. 5, we observe that our INR-based approach achieves superior scores for all metrics. The DIP network [52] performs second best overall, which is consistent with its strong performance on other inverse imaging problems in the computer vision literature. The next best performers are the the gradient descent methods [45], [46] and BREMEN [15]. For the gradient descent methods, we observe that either the total variation or the gradient regularization always slightly outperforms unregularized gradient descent, showing that regularization helps the method output more interpretable structure and edges in the scattering field. BREMEN performs similarly to unregularized gradient descent, which is also expected given that it uses a gradient descent-based optimization to deconvolve the scene.

In Fig. 6, we visualize selected deconvolution results from the experiment summarized in Fig. 5. First, we observe that all methods provide some level of qualitative improvement over DAS reconstruction. We observe that the INR qualitatively reconstructs sharper sand ripples that better match the ground truth image of point scatterers. This performance gap aligns with our quantitative metrics summarized in Fig. 5. Additionally, the INR predictions do not contain many of the circular ringing artifacts that are common to deconvolved images and visible in the competing method’s predictions. Again aligning with quantitative metrics, the DIP network performs closely with the INR, to the point where it is sometimes difficult to qualitatively name a superior method. We observe that the regularized gradient descent methods fail to achieve the level of quality achieved by INR.

\(^3\)We note that we use a unity gain window on the transmit waveform for these experiments to maximize deconvolution opportunity.

\(^4\)We choose this noise range as adding more noise degrades all methods past the point of being qualitatively interpretable.
and DIP, which also aligns with the quantitative results. The BREMEN and Wiener methods contain the most deconvolution artifacts, which is expected since they do not leverage any form of inductive bias like the networks or smoothness regularizers. In other words, these methods do not make any prior assumptions about the scene, such as the scene being comprised of smoothly varying scatterer amplitudes.

We choose not to include the unregularized gradient descent result for brevity’s sake and because it is quantitatively and qualitatively subpar to its regularized counterparts.

**B. Comparative Performance Between Methods**

Our next set of experiments evaluate the performance of all methods with competitive hyperparameter settings. In particular, we consider the average performance of all methods on our simulated dataset at PSNR = 30 dB. Table II summarizes these experiments by providing our three quantitative metrics as well as the seconds per iteration of each method. For each method, we report the best performing PSNR, SSIM, and LPIPS numbers obtained within 5000 iterations.

We observe that our proposed SINR approach, which uses the MLP-based INR, achieves the highest PSNR (20.35) and SSIM (0.72) values. The deep image prior approach, which
uses a bottleneck U-net CNN architecture achieves competitive performance and notably the best LPIPs score (0.15). We observe that an alternative INR architecture, SIREN [61] achieves similar performance to the MLP-based approach. We find that while these methods achieve comparable simulation results, the tunable $\kappa$ parameter enables the MLP-based architecture to achieve superior results on real AirSAS data, a point we discuss more in Section V and Fig. 14.

Next, we highlight that the performance of our SINR approach remains superior to the gradient descent methods regardless of the solver (i.e., SGD versus ADAM). Specifically, the 8 layer MLP-based SINR method achieves 19.38 PSNR with the SGD solver, which is higher than all regularized and unregularized gradient descent methods. Additionally, we observe that reducing the layers from 8 to 2 in the MLP results in a drastic performance regression, suggesting the importance of deep MLP architectures.

Finally, we observe that the gradient descent methods run slightly faster than the neural networks, which is expected given there are fewer steps in the backpropagation graph. However, the BREMEN and Wiener filter algorithms offer the fastest runtimes of all the methods. Additionally, BREMEN achieves impressive PSNR and SSIM scores on this task. We remark that BREMEN outperforms the gradient descent methods on the experiments performed in Table II and underperforms the gradient descent methods on the noise sweep experiments shown in Fig. 6. We use a Tukey window on the transmit waveform for the table experiments and a box window with unity gain for the noise sweep experiments. We conclude that BREMEN potentially performs better when a window is applied to the transmit waveform.

C. Performance Analysis

Our next experiment highlights the advantages of accounting for a spatially-varying CSAS PSF. First, we simulate a scene containing four quadrants of scatterers (Fig. 7), where each quadrant is at a slightly different $z$ depth relative to the transducers. Specifically, consider a modified form of Eq. where $\phi$ is an additional phase shift imparted by scatterers in the scene:

$$S_{T_{\alpha}}(t) = \int \int \sigma(x, y, z_0) \cdot w \left( t - 2 \frac{||T_{\alpha} - (x, y, z_0)||}{c} + \phi \right) \, dx \, dy + n(t).$$

(15)

We simulate this phase shift such that scatterers within the top right and bottom left quadrants contain strictly real components, while the top left and bottom right quadrants contain strictly imaginary components. We then show deconvolution results using our INR under two scenarios. First, we adopt the strategy of

| Meth. | Opt. | LR | Other Details | seconds/iteration | PSNR ↑ | SSIM ↑ | LPIPS ↓ |
|-------|------|----|---------------|------------------|-------|-------|--------|
| SINR  | ADAM | 1e-5 | 8 layers, $\kappa = 1$ | .26 | 11.48 | 0.35 | 0.87 |
|       |      |     | 8 layers, $\kappa = 10$ |     | 20.35 | 0.72 | 0.20 |
|       |      |     | 8 layers, $\kappa = 20$ |     | 18.13 | 0.63 | 0.21 |
|       |      |     | 8 layers, $\kappa = 40$ |     | 15.41 | 0.47 | 0.46 |
|       |      | 1e-4 | 2 layers | .17 | 15.34 | 0.48 | 0.39 |
|       | SGD  | 1e-5 | SIREN | .19 | 20.30 | 0.72 | 0.17 |
|       |      | 1e-4 | 8 layers | .25 | 19.38 | 0.71 | 0.37 |
| GD    | SGD  | 1e-5 | n/a | .16 | 17.91 | 0.67 | 0.41 |
|       |      | 1e-3 | 10. |     | 18.50 | 0.68 | 0.37 |
|       |      | 1e-4 | 10. |     | 18.58 | 0.68 | 0.36 |
|       |      | 1e-5 | 10. |     | 18.35 | 0.68 | 0.36 |
|       |      | 1e-1 | 10. |     | 18.82 | 0.65 | 0.48 |
|       |      | 1.   | 10. |     | 9.10 | 0.33 | 0.77 |
| GD + TV | SGD | 1e-3 | $\beta = 1e-7$ | .16 | 15.01 | 0.59 | 0.48 |
|        |      |     | $\beta = 1e-8$ |     | 18.65 | 0.67 | 0.35 |
|        |      |     | $\beta = 1e-9$ |     | 18.64 | 0.68 | 0.36 |
|        |      |     | $\beta = 1e-10$ |    | 18.59 | 0.68 | 0.36 |
| GD + Grad. Reg | SGD | 1e-3 | $\beta = 1e-9$ | .16 | 17.67 | 0.68 | 0.38 |
|        |      |     | $\beta = 1e-10$ |    | 18.37 | 0.68 | 0.36 |
|        |      |     | $\beta = 1e-11$ |   | 18.56 | 0.68 | 0.36 |
| DIP   | ADAM | 1e-5 | n/a | .52 | 19.09 | 0.69 | 0.15 |
|       |      | 1e-3 | 10. |     | 20.17 | 0.71 | 0.18 |
|       |      | 1e-2 | 10. |     | 20.04 | 0.71 | 0.18 |
|       |      | 1e-1 | 10. |     | 19.05 | 0.67 | 0.20 |
| BREMEN | n/a | - | n/a | .03 | 18.66 | 0.70 | 0.41 |
|       | Wiener | n/a | $\alpha = 1e-4$ |   | 14.00 | 0.38 | 0.46 |
|       |      | - | $\alpha = 1e-3$ |   | 15.63 | 0.44 | 0.35 |
|       |      | - | $\alpha = 1e-2$ |   | 14.98 | 0.42 | 0.44 |

All methods were run up to 5000 iterations for comparison. The columns from left to right specify: (1) the solver used (i.e., SGD versus ADAM), (2) the learning rate (LR), (3) any important details like architecture settings or regularization parameters, (4) the number of seconds per iteration, (5) and quantitative metrics. The highest achieved quantitative metrics are bolded.
BREMEN [15] by deconvolving the scene assuming a spatially invariant centered PSF. This yields a deconvolution result where the strictly real quadrants are correctly deconvolved, but the strictly imaginary components contain significant artifacts. This is expected, as the centered PSF is real — convolving a real PSF with real scene scatterers yields a real DAS estimate. Thus, the forward model has no method for matching the imaginary terms in the DAS reconstruction.

Next, we perform an experiment that highlights a limitation of our deconvolution method. In our model, we assume that all points scatter acoustic energy omnidirectionally. Specifically, a scatterer returns the same energy regardless of view angle. We ignore multiple scattering and elastic effects (where the scatterer absorbs and later re-radiates incident energy). These effects occur due to target geometry and material properties, but their magnitude is typically small compared to the omnidirectional scattering that we model. To characterize our performance in the presence of these second-order effects, consider measurements from (15) where each scatterer is assigned a random phase within \( [0, 2\pi] \) (i.e., \( \Phi(x, y, z) \)) to simulate a diffuse scattering field. We simulate such a scene in Fig. 8 and show that the deconvolution result is only marginally superior to the DAS reconstruction. In real data, this situation is akin to a speckle field, for example one generated from a sandy ocean floor with virtually no structure except for small and random height fluctuations in each particle of sand. In such a case, the underlying scattering phase lacks structure making coherent deconvolution challenging. On the other hand, we show that our convolution method succeeds in regions of structured (i.e. uniform) phase. To highlight this fact, we simulate scatterers in a grid (bottom row of Fig. 8) where each grid cell contains a uniform phase within 0 and 2\( \pi \). While the image contains some edge artifacts where the phase regions meet, we show that our deconvolution enhances the image features within each grid cell and achieves a substantially higher PSNR score.

Fig. 7. Importance of Phase in Coherent Deconvolution: The top left image shows ground truth reflectors that impart an additional phase shift (white text) on the incident signal. The top right image shows the DAS reconstruction of the scene and each quadrants PSNR (white text) relative to ground truth. Deconvolution methods that assume a spatially invariant PSF and don’t account for phase fail to deconvolve regions of the scene where the reflectors effect an incoherent phase shift on the incident signal (bottom left). Our approach is able to handle this issue by predicting a complex scene that, when convolved with the centered PSF, fits the real and imaginary components of the DAS image and correctly deconvolves all four quadrants (bottom right).

V. REAL EXPERIMENTAL RESULTS

This section discusses deconvolution results on AirSAS [16] data. The ideal deconvolution method should recover a sharp outline of the object, highlighting the spatial locations where acoustic energy was backscattered to the microphone, and attenuate the side-lobe energy artifacts caused by the finite bandwidth of the PSF. We run methods for a fixed number of iterations until the deconvolution results are qualitatively converged. Additionally, we tune all hyper-parameters to encourage the best performance of each method.

Experimental data was collected using an in-air circular synthetic experiment housed in an anechoic chamber, as shown in Fig. 9 [16]. We build this setup in-air because it allows for experimental control that is otherwise impossible or expensive to achieve in water. Since we are interested in the response from a rigid, impenetrable target in this work, the relevant physics is directly analogous between air and water. The targets were centered on a 0.2 m turntable and rotated in 1° increments relative to an transducer array consisting of loudspeaker tweeter (Peerless OX20SC02-04) and a microphone (GRAS 46AM)
0.85 meters away and 0.25 meters above the scene center. The tweeter transmits a linear frequency modulated chirp for a duration of 1 ms waveform at center frequency $f_c = 20$ kHz and bandwidth $\Delta f = 20$ kHz or $\Delta f = 5$ kHz depending on the experiment. The microphone detects backscattered acoustic energy from the target. In this work, we transmit LFMs because they offer an attractive trade-off between spatial resolution and requiring a low peak to average power ratio that enables them to be reproduced linearly at high power by transducers with finite displacements.

### A. Main Deconvolution Results

Fig. 10 shows ours and competing deconvolution results on three scenes from AirSAS. We first highlight Fig. 10(a), and observe that the INR and DIP methods are the only methods to accurately recover the shape of the peninsula on the cork-cutout (left inlet), and generally outperform all other methods at recovering the object geometry. We observe that the DIP network suffers from low-frequency noise artifacts that we believe stem from its inability to fit the high frequency point scatterers in our model. Note that this artifact is not observed in DIP’s results on simulated scenes, and thus highlights the importance of validating our deconvolution methods on real acoustic data captured with AirSAS. All other methods seem to improve performance over DAS reconstruction, but fail to match the performance of the networks. We observe that BREMEN fails to capture the peninsula on the cork-cutout since this feature falls within the incoherent (i.e., imaginary) part of the measurements, which BREMEN is unable to account for. We also highlight Fig. 10(c), and note that even at 30 dB, the INR attenuates noise while recovering the scene’s salient features, particularly around the scissors handles.

Our next AirSAS result is shown in Fig. 11, where we task methods with deconvolving images of the same object shown in...
Fig. 11. Low Bandwidth Deconvolution: Example of the scene shown Fig. 10(a), captured with a low bandwidth 5 kHz waveform. Our method (INR) drastically outperforms other methods at recovering the object’s edges.

Fig. 10, but captured with a lower-bandwidth, 5 kHz waveform. We observe that the DAS reconstruction significantly distorts the object geometry due to the high side-lobe energy in the matched-filtered waveform. Impressively, our INR approach is the only method to recover the objects finer features (for example the peninsula shown on the left inlet). The competing methods do make improvements over the DAS reconstruction, but do not recover salient object edges like the INR.

B. Performance Analysis

In Fig. 12, we provide a qualitative example of the advantages the inductive bias of an INR provides over the gradient descent methods. Here, we show the DAS target, the estimated DAS images synthesized from convolving the INR and gradient descent outputs with the PSF, and the deconvolved results at different iterations during the method’s respective optimizations. While both methods fit the DAS target well, the predicted deconvolutions are different. The INR is beginning to recover the object edges at iteration 50, while the gradient descent result contains significant blurring at iteration 10,000. This provides an example of the advantage gained by leveraging an INR’s inductive bias.

In Fig. 13, we show that the ability for the INR to fit to high frequency details of the point scatterers is evidenced by the choice of its bandwidth parameter $\kappa$. From the figure, we observe that setting the bandwidth too low results in reconstructions similar to the DIP model, as the network is unable to fit the high frequency details in the point scattering model. Additionally, we observe that the choice of the learning rates influences the reconstruction quality, which demonstrates the importance of tuning the model hyperparameters to maximize performance.

Finally, we consider the comparison between an INR architecture constructed with fully connected layers and ReLU activations (the method we recommend for implementing SINR and referred to as MLP + ReLU), and the SIREN architecture [61]. We observe that both methods achieve comparable results on the
Fig. 13. INR Hyperparameter Sweep: Our INR’s reconstruction results of the same 5 kHz measured scene shown in 12 for varying INR bandwidth parameters \( \kappa \) and learning rates. We observe that setting \( \kappa \) too low yields reconstructions similar to DIP, where the INR unable to fit high frequency content of our point scattering model. Conversely, setting \( \kappa \) too high causes the network to choose too sparse a solution or overfit to noise in the scene. For this scene, the optimal bandwidth and learning rate is 30 and 0.001, respectively, since these settings recover salient scene features while suppressing noise.

Fig. 14. INR Architecture Comparison: We observe that an INR constructed from fully connected layers and ReLU activations (MLP + ReLU) yields superior reconstructions to the SIREN architectures which does not explicitly use a positional encoding parameter [61].

simulated scenes. However, one real scenes, we observe that the SIREN’s lack of a tunable positional encoding parameter results in inferior reconstructions. In Fig. 14, we show reconstruction results for the small feature cutout scene using a \( \Delta f = 20 \) kHz waveform. By tuning the positional encoding parameter of the MLP+ReLU architecture, we achieve a deconvolved scene that recovers the outline of the object and attenuates background noise. The SIREN-based reconstruction recovers a less detailed object outline and amplifies background noise.

VI. DISCUSSION

We propose a CSAS deconvolution method, SINR, that leverages the inductive bias of INRs to enhance reconstructed imagery. We demonstrate though simulated and real results that our method outperforms competing deconvolution methods. This capability has potential for enhancing existing target recognition and underwater visualization tasks, where limited resolution makes it challenging for man or machine to determine salient features of a target.

There are limitations of our approach which provide avenues for future work. In the 2D case used in this paper, we show that it is important for our deconvolution method to account for a spatially-varying CSAS PSF. We address this problem by tasking a neural network with predicting a complex field, that, when convolved with the centered PSF, is capable of producing a complex DAS estimate that can fit the given DAS reconstruction. While we demonstrate this strategy yields advantages, it casts the PSF’s spatially varying nature to the weights of a network, which limits interpretablity. Future work could investigate methods to efficiently model the PSF’s spatially varying properties while deconvolving scenes, which may enhance deconvolution results.

Neural networks are black boxes by nature. While our INR can achieve high quality reconstructions, the wrong choice of hyperparameters can yield substantial reconstruction artifacts. Further, these artifacts are only obvious in hindsight, which can make deep learning methods difficult to deploy in critical situations. However, uncertainty quantification in neural networks is an active area of research [75], and future work should investigate methods for enabling a network to accurately quantify its reconstruction certainty.

Our point scattering model makes several assumptions about the scene to be more computationally tractable. First, we assume an omnidirectional scattering model and we do not account for occlusion. It remains an interesting direction of future work to account for anisotropic scatterers and occlusion, perhaps by adding a dependence on look angle to the network. Accounting for these effects may enhance results, particularly for 3D reconstructions.

Regarding the realism of the AirSAS system, sonar measurements captured in-air serve as a valuable proxy for in-water measurements. However, imaging underwater must address additional challenges such as non-homogeneous mediums and sensor location uncertainty. Future work should seek to evaluate the performance of the proposed method under these conditions. Additionally, future work should investigate the application of our deconvolution method to 3D SAS data where a CSAS senses depth by hovering above the seafloor at different heights or radii. Finally, we hope to extend this method to additional geometries, like side-scan SAS, in future work.

6The INR result in Fig. 11 is marginally superior because we run with slightly different learning rate and for more iterations.
Nevertheless, we believe the proposed method pushes CSAS imaging capabilities a step further, particularly by enabling low bandwidth systems to recover salient scene features that are degraded with conventional reconstruction methods. We hope this work sparks additional interest in applying neural networks to SAS image reconstruction and proves useful for future imaging tasks.

ACKNOWLEDGMENT

The authors acknowledge Research Computing at Arizona State University for providing GPU resources that have contributed to the research results reported within this paper.

REFERENCES

[1] A. Bellettini and M. Pinto, “Design and experimental results of a 300-kHz synthetic aperture sonar optimized for shallow-water operations,” IEEE J. Ocean. Eng., vol. 34, no. 3, pp. 285–293, Jul. 2009.
[2] M. P. Hayes and P. T. Gough, “Synthetic aperture sonar: A review of current status,” IEEE J. Ocean. Eng., vol. 34, no. 3, pp. 207–214, Jul. 2009.
[3] R. E. Hansen, “Introduction to synthetic aperture sonar.” Accessed: Mar. 15, 2022. [Online]. Available: www.intechopen.com
[4] H. J. Callow, “Signal processing for synthetic aperture sonar image enhancement,” Ph.D. dissertation, Univ. Canterbury, Christchurch, New Zealand, 2003.
[5] M. P. Hayes and P. T. Gough, “Broad-band synthetic aperture sonar,” IEEE J. Ocean. Eng., vol. 17, no. 1, pp. 80–94, Jan. 1992.
[6] I. D. Gerg et al., “GPU acceleration for synthetic aperture sonar image reconstruction,” in Proc. Glob. Oceans Singap.–US Gulf Coast, 2020, pp. 1–9.
[7] D. S. Flotnick, P. L. Marston, and T. M. Marston, “Fast nearfield to farfield conversion algorithm for circular synthetic aperture sonar,” J. Acoust. Soc. Amer., vol. 136, no. 2, pp. EL61–EL66, 2014.
[8] T. M. Marston and J. L. Kennedy, “Volumetric acoustic imaging via circular multipass aperture synthesis,” IEEE J. Ocean. Eng., vol. 41, no. 4, pp. 852–867, Oct. 2016.
[9] D. P. Williams, “Underwater target classification in synthetic aperture sonar imagery using deep convolutional neural networks,” in Proc. 23rd Int. Conf. Pattern Recognit., 2016, pp. 2497–2502.
[10] N. Nadimi, R. Javidan, and K. Layeghi, “Efficient detection of underwater natural gas pipeline leak based on synthetic aperture sonar (SAS) systems,” J. Mar. Sci. Eng., vol. 9, no. 11, 2021, Art. no. 1273.
[11] Ø. Sture, M. Ludvigsen, M. S. Scheide, and T. Thorlsen, “Recognition of cold-water corals in synthetic aperture sonar imagery,” in Proc. IEEE/OES Auton. Underwater Veh. Workshop, 2018, pp. 1–6.
[12] P. T. Gough and D. W. Hawkins, “Unified framework for modern synthetic aperture imaging algorithms,” Int. J. Imag. Syst. Technol., vol. 8, no. 4, pp. 343–358, 1997.
[13] Y. Paillais, Y. Petitblot, and B. Mulgrew, “Increasing circular synthetic aperture sonar resolution via adapted wave atoms deconvolution,” J. Acoust. Soc. Amer., vol. 141, pp. 2623–2632, 2017.
[14] Y. Paillais, “2D & 3D, centred & offset, circular synthetic aperture sonar point spread function,” in Proc. Conf. UACOE, 2019, pp. 1–8.
[15] P. de Heering, K. U. Simmer, E. Ochieng-Ogolla, and A. Wasiljeff, “A deconvolution algorithm for broadband synthetic aperture data processing,” IEEE J. Ocean. Eng., vol. 19, no. 1, pp. 73–83, Jan. 1994.
[16] T. E. Blanford, J. D. McKay, D. C. Brown, J. D. Park, and S. F. John-son, “Development of an in-air circular synthetic aperture sonar system as an educational tool,” in Proc. Meetings Acoust., 2019, vol. 36, Art. no. 070002.
[17] C. H. Sherman and J. L. Butler, Transducers and Arrays for Underwater Sound, 2nd ed. Berlin, Germany: Springer, 2016.
[18] B. G. Ferguson and R. J. Wyber, “Generalized framework for real-time, synthetic aperture, and tomographic sonar imaging,” IEEE J. Ocean. Eng., vol. 34, no. 3, pp. 225–238, Jul. 2009.
[19] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, “Image deblurring with blurred/noisy image Pairs,” in Proc. ACM SIGGRAPH Papers, 2007, pp. 1–8.
[20] J. Biemond, R. L. Lagendijk, and R. M. Mersereau, “Iterative methods for image deblurring,” Proc. IEEE, vol. 78, no. 5, pp. 856–883, May 1990.
[21] R. Wang and D. Tao, “Recent progress in image deblurring,” 2014, arXiv:1409.6838.
T. F. Brooks and W. M. Humphreys, “A deconvolution approach for the mapping of acoustic sources (DAMAS) determined from phased microphone arrays,” J. Sound Vib., vol. 294, no. 4/5, pp. 856–879, 2006.

R. Dougherty, “Extensions of damas and benefits and limitations of deconvolution in beamforming,” in Proc. 11th AIAA/CEAS Aeroacoustics Conf., 2005, Art. no. 2961.

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

K. Gong, C. Catana, J. Qi, and Q. Li, “PET image reconstruction using deep image prior,” IEEE Trans. Med. Imag., vol. 38, no. 7, pp. 1655–1665, Jul. 2019.

D. O. Baguer, J. Leuschner, and M. Schmidt, “Computed tomography reconstruction using deep image prior and learned reconstruction methods,” Inverse Problems, vol. 36, no. 9, 2020, Art. no. 094004.

D. V. Veen, A. Jalal, M. Soltanolkotabi, E. Price, S. Vishwanath, and A. G. Dimakis, “Compressed sensing with deep image prior and learned regularization,” 2020. [Online]. Available: https://openreview.net/forum?id=Hkl-4r8t4r

B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “NERF: Representing scenes as neural radiance fields for view synthesis,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 405–421.

Y. Xie et al., “Neural fields in visual computing and beyond,” 2021. [Online]. Available: https://neuralfields.cs.brown.edu/

B. Neyshabur, R. Tomioka, and N. Srebro, “In search of the real inductive bias,” in Proc. 3rd Int. Conf. Learn. Representations, B. Yoshua and L. Yann, Eds., San Diego, CA, USA, May 2015, [Online]. Available: http://arxiv.org/abs/1412.6614

A. Bietti and J. Mairal, “On the inductive bias of neural tangent kernels,” in Proc. Adv. Neural Inf. Process. Syst., 2019, vol. 32, pp. 12893–12904.

V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, “Implicit neural representations with periodic activation functions,” in Proc. Adv. Neural Inf. Process. Syst., 2020, vol. 33, pp. 7462–7473.

W. Shen et al., “Comparison of coherent and incoherent beamforming envelope detectors for norsar regional seismic events,” Texas Instur. Inc Dallas Equipment Group, Dallas, TX, USA, Tech. Rep. 6, 1974.

J. G. Nagy and D. P. O’Leary, “Fast iterative image restoration with a spatially varying PSF.” Proc. SPIE, vol. 3162, pp. 388–399, 1997.

J. G. Nagy and D. P. O’Leary, “Restoring images degraded by spatially variant blur.” SIAM J. Sci. Comput., vol. 19, no. 4, pp. 1063–1082, 1998.

D. Azinovic, T.-M. Li, A. Kaplanly, and M. Nielbner, “Inverse path tracing for joint material and lighting estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2447–2456.

C.-Y. Tsai, A. C. Sankaranarayanan, and I. Gkioulekas, “Beyond volumetric albedo–A surface optimization framework for non-line-of-sight imaging,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 1545–1555.

A. F. Agarap, “Deep learning using rectified linear units (RELU),” 2018, arXiv:1803.08375.

“Nvidia a100 gpus power the modern data center.” Accessed: Oct. 1, 2022. [Online]. Available: https://www.nvidia.com/en-us/data-center/a100/

W. Shen, “Extrapolation, Interpolation, and Smoothing of Stationary Time Series.” Cambridge, MA, USA: MIT Press, 1964.

R. W. Schafer, R. M. Mersereau, and M. A. Richards, “Constrained iterative restoration algorithms,” Proc. IEEE, vol. 69, no. 4, pp. 432–450, Apr. 1981.

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 586–595.

J. Zhang, J. Pan, W.-S. Lai, R. W. Lau, and M.-H. Yang, “Learning fully convolutional networks for iterative non-blind deconvolution,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 3817–3825.

I. D. Gerg and D. P. Williams, “Additional representations for improving synthetic aperture sonar classification using convolutional neural networks,” in Proc. Inst. Acoust., 2018, vol. 40.

Y. Xue, S. Cheng, Y. Li, and L. Tian, “Reliable deep-learning-based phase imaging with uncertainty quantification,” Optica, vol. 6, no. 5, pp. 618–629, 2019.