INSAS: Implicit Neural Representations for Synthetic Aperture Sonar

EEE 598 - Final Project Paper. Predicting Missing SAS Measurement with Implicit Neural Representations.

Ali Almuallem[1]
School of Computing and Augmented Intelligence; Arizona State University[1]

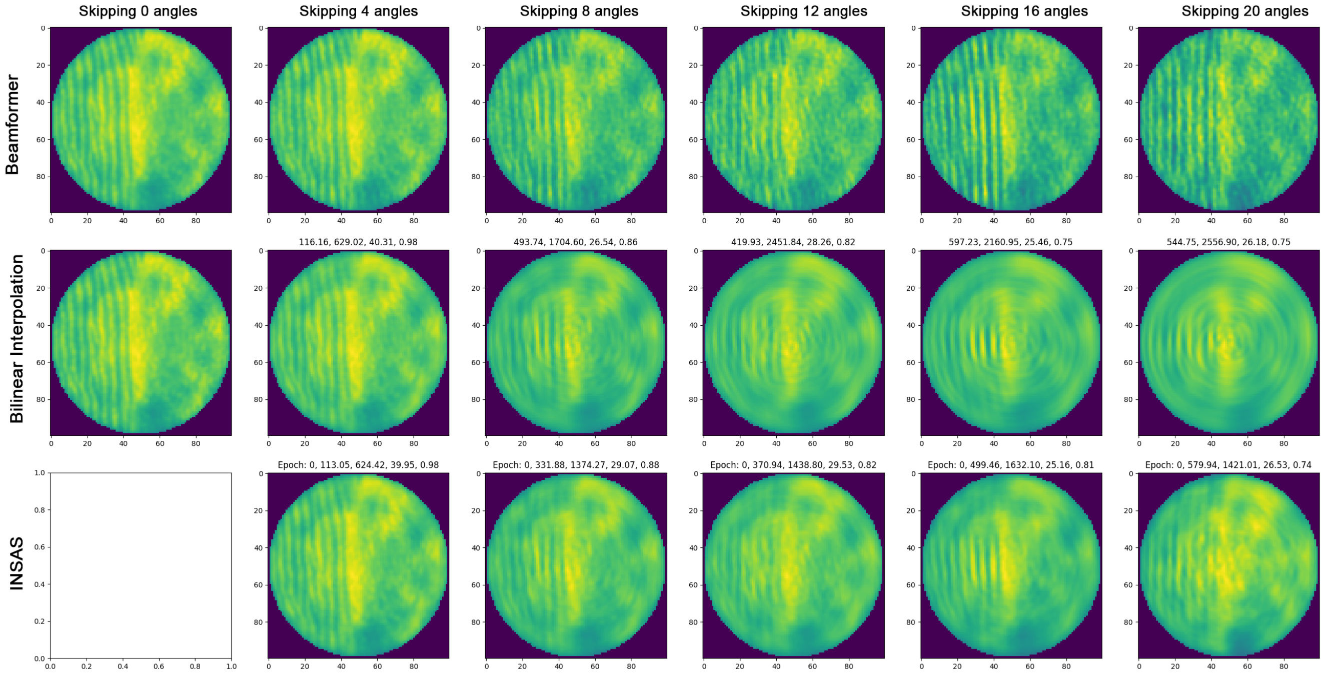

Figure1. The first, second, and third row represent beamformer, bilinear interpolation, and *INSAS* results, respectively. From left to right, the results of skipping 0, 4, 8, 12, 16, and 20 angles, respectively.

***Abstract*—Circular Synthetic Aperture Sonar (CSAS) takes a series of acoustic measurements, usually underwater, to construct an image of a scene. The motivation for using Synthetic Aperture Sonar (SAS) over conventional optical cameras is that SAS can capture images of scenes far away, like a seafloor several tens to hundreds of meters away from the receiver. SAS can also penetrate objects and provide an idea of the depth of an object in some cases. Hence, it is commonly used in the search for wreckage. [1]**

Our work utilizes Implicit Neural Representations (INR) to reconstruct CSAS images from sub-sampled measurements. Our INR for SAS architecture, which we call *INSAS*, competes against conventional beamforming, and bilinear interpolation methods and provides a continuous representation from sparse discrete measurements. To encourage the reproducibility and usability of *INSAS*, we will publicly provide the code and datasets.

***Index Terms*—sonar, reconstruction, interpolation**

I. Introduction

CSAS is an important imaging modality that captures seafloors using acoustics. However, just like other modalities that combine multiple measurements or projections to construct an image, SAS quality is bounded by the number of those measurements. Down-sampled SAS (smaller number of measurements) produces visual artifacts and aliasing. If a vessel has to move faster and takes fewer samples, or if some measurements were corrupted, there is no proposed method -based on our knowledge- to account for those missing / corrupted measurements. Our work investigated the possibility of using INRs to interpolate those missing measurements. INRs learn a continuous function of an underlying sparse signal and have shown promising results in other domains [2]

INSAS takes a 2-D coordinate mesh of angles and samples as an input to the neural network. The network is then tasked to map those coordinates with their respective intensity values. We will train the network on a uniformly down-sampled coordinate mesh to account for the missing angles scenario. For example, if the original measurement was taken from 360 angles, we could down-sample the input by a factor of

4 by feeding the network each $4 * n$ angle, where n is the maximum angle and skipping the rest. The angle part of the mesh would have a sequence of this form $[0, 4, 8, ..., n]$. Each of these angles corresponds to a slightly different part of the scene; however, they may share similar spatial and perceptual features. Our *INSAS* network, in this case, tries to learn a continuous function between those angles. The continuous representation learned by *INSAS* emphasizes the continuum nature of any scene taken in nature.

Our work could inspire new technological advancements and solve some existing obstacles with SAS. Being able to drop some angles and still get good-quality SAS images may enable ships and unmanned surface vessels (UAVs) to move faster. *INSAS* could also mitigate cases where a signal from one or few angle measurements is corrupted.

In our work, we acknowledge that according to our best knowledge, there was no prior work that used INRs to interpolate missing angular measurements in CSAS. We can summarize our contributions to the following:

1) We proposed a pipeline using implicit neural representation to interpolate missing angular measurements in CSAS.
2) We benchmarked our results against two conventional approaches and report the quantitative and qualitative results.
3) We analyzed our pipeline with different simulated scenes and real data and drew useful empirical conclusions that would inspire the field.
4) We will release the simulated and real dataset along with the code for the public.

II. Related Work

A. Implicit Neural Representations and Neural Radiance Fields

Our *INSAS* pipeline leverages implicit neural representations (INR) as a building block to build a learning-based interpolation method. INRs, also sometimes referred to as Coordinate Based Networks (CBN), take a coordinate mesh as input and map it to an underlying signal of interest. In 2-D images, for example, the network input is a 2-D grid of (x, y) coordinates, and the signal is RGB values or grayscale intensities. In our pipeline, we encode the angular coordinates $[0, 4, 8, ...360]$ and the samples collected per angle $[0, 1, 2, ..., 1000]$ and map them to the corresponding ground truth intensity values.

The architecture is usually constructed from a fully connected, multi-layer perceptron network (MLP) with an activation function between each layer.

INRs strength comes in the fact that they learn a continuous function from discrete, low-dimensional input. Therefore, the applications to INR are vast and range from representation [2] [3], reconstruction from limited views [4] [5] [6], to compression [7], and many more.

Despite their potential, vanilla INRs suffer when learning to represent high frequencies. Some solutions to this problem have been proposed and adopted widely and are discussed below. Another drawback of INRs is their volatility when changing the underlying signal. Some hyperparameters work best with specific scenes but not others. We hypothesize that this could be due to the varying frequencies represented in each scene. Some scenes include a lot of fine details that yield high frequencies. Others have much less so. A solid analysis and framework to mitigate this issue are, based on our knowledge, yet to be proposed.

B. Fourier Features Positional Encoding

To overcome the INRs' inability to learn high-frequency details, a positional encoding was proposed. Known as Fourier features [8], this positional encoding is composed of random sinusoidal vectors the input gets multiplied by. The premise is to encode the low dimensional input coordinates in a Fourier-like manner that enables the network to pick up the high frequencies.

Fourier features, however, are not a perfect solution yet. To account for the varying frequencies from one scene to another, it is proposed to multiply those features by a scalar that would represent the bandwidth. Hence, the results of using Fourier features vary slightly based on the architecture and the scalar value used to multiply those features. This represents an obstacle and a slight inconvenience when prototyping an INR on new scenes since there is usually no knowledge about the frequencies represented in the scene.

C. INR for Deconvolving SAS Images

In practice, SAS systems have limited bandwidth. Often, this bandwidth is not high enough to capture the high-frequency details of a scene. This results in blurry SAS images. Deconvolving the resulting blurry SAS images with the scene point spread function (PSF) should yield, in theory, sharper images. However, since deconvolution is an inverse operation, it is a highly-ill posed noise-prone problem. [9]

One proposed solution [9] is to obtain a PSF of one point scatterer placed at the center of the scene. Deconvoving with this PSF could suppress the reconstruction artifacts caused by side lobes. It is worth noting, however, that this method assumes an invariant PSF across the scene, a condition that does not hold in reality.

As INRs became faster and easier to construct and run, an extension of this work may experiment with multiple PSFs to account for the variations across the scene.

III. Method

Our pipeline relies on a learning-based approach. We leverage INRs as function approximators. We discuss the neural network architecture choice, the criteria for choosing the best results, and any assumptions in this section.

A. Simulated SAS Measurements

INSAS has been trained extensively on simulated SAS data generated by a ray-based method. The quantitative results of the simulated data are averaged over five different simulated scenes, as shown in figure XXXXYYYY.

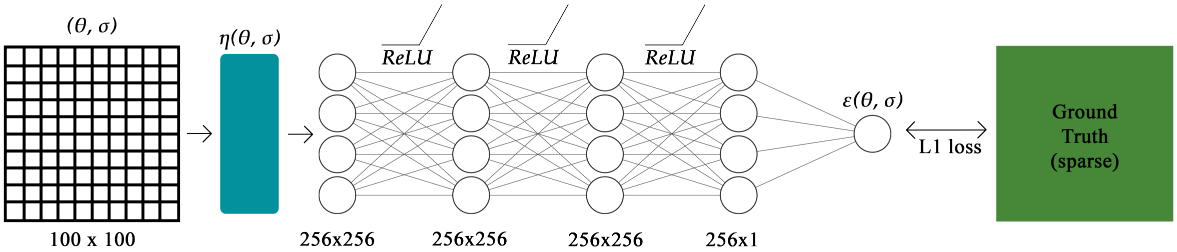

Fig. 1. *INSAS* is constructed from a fully-connected neural network. The input is a coordinate mesh of angles and samples. The loss is computed using L1 loss.

The simulated scenes are taken using a circular array geometry. The scenes are 0.2×0.2 meters, with the transducers circulating radially 0.85 meters away from the center of the scene with a height of around 0.25 meters. This setup closely simulates the AirSAS setup. All scenes are 100×100 pixels.

B. Real Data: AirSAS Measurements

To confirm the real-life applicability of our *INSAS* pipeline, we demonstrate its performance on real data captured by a circular air-SAS system called AirSAS [10]. The data was captured by [11] using a rotating turntable coupled with a fixed microphone and a tweeter, effectively mimicking a CSAS system.

C. Neural Network Architecture

To learn a continuous function of the sparse from sub-sampled measurements that are angularly sparsed, we leverage a special type of neural network known as Implicit Neural Representation (INR). An INR is commonly constructed out of a fully-connected neural network that takes in a low-dimensional input and outputs a signal of interest.

Formally, we use an INR to map the SAS scene measurements (θ, σ) to the real part of a complex waveform associated with these coordinates. Here, θ represents angles from 0°to 360°, and σ represents an array of 1000 samples for each angle. Those samples represent the measured pressure by the SAS microphone for the associated angle. $F_\phi : (\theta, \sigma) \mapsto \epsilon(\theta, \sigma)$ where $\epsilon(\theta, \sigma)$ is the real value part of a complex waveform at the (θ, σ), and ϕ are the trainable parameters of the network. The optimization is calculated using an L1 loss as follows:

$$\min_{\phi} ||b_{estimated}(\theta, \sigma) - b_{truth}(\theta, \sigma)||_1 \tag{1}$$

Where $b_{estimated}(\theta, \sigma)$ are the estimated values of the real part given by the neural network and $b_{truth}(\theta, \sigma)$ are the ground truth waveform real values associated with those angular measurements and samples. Since the network is learning on sub-sampled angular measurements, the $b_{estimated}(\theta, \sigma)$ and $b_{estimated}(\theta, \sigma)$ contain a sub-sample of the original 360°angles. We conducted multiple experiments with different angular sparsity, which we report in the later sections.

The input to the network is a sub-sample angular mesh. In the case of skipping three consecutive angular measurements, the input mesh to our network is:

$$\begin{bmatrix} angle0_{sample0} & angle0_{sample1} & ... & angle0_{sample999} \\ angle4_{sample0} & angle4_{sample1} & ... & angle4_{sample999} \\ angle8_{sample0} & angle8_{sample1} & ... & angle8_{sample999} \\ . & . & ... & . \\ . & . & ... & . \\ angle352_{sample0} & angle352_{sample1} & ... & angle352_{sample999} \\ angle356_{sample0} & angle356_{sample1} & ... & angle356_{sample999} \end{bmatrix}$$

Where each column σ represents the n_{th} sample at the m_{th} angle θ, and the output of the network is the real part of the waveform associated with those angles and samples.

Our network learns using an analysis-by-synthesis approach, and therefore it is self-supervised and does not need training data. We constructed the network of three fully-connected layers with ReLU activation function in between.

As INRs struggle to learn high-frequency signals, we positionally encoded our input mesh in a random Fourier features latent vector that enables the network to represent fine spatial details [8]. The positionally encoded input $\eta(\theta, \sigma)$, along with the whole neural network architecture, are shown in Figure 1.

IV. Variation Regularizer

Toward achieving our task of learning the underlying continuous function from sparse measurements, we hypothesized that a variation regularizer between each two sparse angular measurements might produce better SAS images. We formulated our hypothesis as follows:

$$reg(x) = (S^2(x) - S^2(x-1))^2 + (S^2(x) - S^2(x+1))^2 \tag{2}$$

where $reg(x)$ is the regularizer term at the angular measurement x, and $S^2(x)$ is the sample variance at this angular measurement. Equation 3 shows how the sample variance, $S^2(x)$, is calculated.

$$S^2(x) = \frac{\sum_{i=0}^{n}(x_i - \bar{x})^2}{n-1} \tag{3}$$

$$\min_{\phi}(||b_{estimated}(\theta, \sigma) - b_{truth}(\theta, \sigma)||_1 + \lambda * reg) \tag{4}$$

TABLE I
RESULTS WITH VARIATION REGULARIZER. AVERAGE PSNR, SSIM, AND PERCEPTUAL LOSS (LPIPS) BY ANGLE STEP OVER FIVE SIMULATED SCENES. BETTER RESULTS ARE IN BOLDFACE

Method/Angle step	**4**	**8**	**12**	**16**	**20**
	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS
Beamformer	**41.469**, 0.977, 0.01	31.003, 0.892, 0.041	28.722, 0.838, 0.06	25.844, 0.781, 0.089	22.916, 0.728, 0.109
Bilinear	39.776, 0.976, 0.015	30.384, 0.87, 0.085	29.891, 0.828, 0.107	27.711, 0.764, 0.134	26.273, 0.754, 0.147
INSAS	41.307, **0.979**, **0.009**	**33.844**, **0.898**, **0.035**	**31.927**, **0.853**, **0.047**	**29.052**, **0.820**, **0.052**	**27.018**, **0.778**, **0.058**

TABLE II
RESULTS WITHOUT VARIATION REGULARIZER. AVERAGE PSNR, SSIM, AND PERCEPTUAL LOSS (LPIPS) BY ANGLE STEP OVER FIVE SIMULATED SCENES. BETTER RESULTS ARE IN BOLDFACE

Method/Angle step	**4**	**8**	**12**	**16**	**20**
	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS	PSNR, SSIM, LPIPS
Beamformer	**41.469**, 0.977, **0.009**	31.002, 0.892, 0.041	28.723, 0.838, 0.060	25.846, 0.782, 0.089	22.914, 0.728, 0.108
Bilinear	39.777, 0.975, 0.015	30.384, 0.869, 0.086	29.892, 0.827, 0.108	27.711, 0.764, 0.132	26.272, 0.754, 0.147
INSAS	41.205, **0.978**, 0.01	**33.931**, **0.898**, **0.036**	**31.956**, **0.853**, **0.046**	**29.116**, **0.82**, **0.053**	**26.935**, **0.777**, **0.058**

We incorporated this regularizer term in our loss function as shown in equation 4, where λ is a tunable weight we experimented with from the following values: [0, 0.2, 0.4, 0.6, 0.8, 1]. We report our findings in the results section.

V. RESULTS

Over five simulated scenes, our pipelines produced better quantitative results than both the standalone beamformer and the bilinear interpolation method. Our pipeline is yet to be tested and verified on real data. More analysis will be given as soon as more results are gathered.

TABLE III
AVERAGE METRICS OVER 5 SIMULATED SCENES - WITHOUT A REGULARIZER

Method	**Metrics**		
	PSNR↑	***SSIM***↑	***LPIPS***↓
Beamformer	29.9909	0.8433	0.0618
Bilinear	30.8070	0.8384	0.0978
INSAS	**32.6283**	**0.8655**	**0.0400**

The results in Table III show the peak-signal-to-noise-ratio (PSNR), structural similarity index measure (SSIM), and perceptual loss (LPIPS) [12]. The results were averages over five different simulated scenes and over five angle steps scenarios per scene [4, 8, 12, 16, 20]. We show the results per each angle-step scenario in table II.

Our variation regularizer, contrary to our hypothesis, did not produce a noticeable improvement over our original loss function detailed in equation 1. The average results of our *INSAS* pipeline with the variation regularizer included are detailed in table IV. These results are averaged over five different simulated scenes and over five angle steps scenarios per scene [4, 8, 12, 16, 20]. We show the results per each angle-step scenario in table I.

TABLE IV
AVERAGE METRICS OVER 5 SIMULATED SCENES - WITH REGULARIZER

Method	**Metrics**		
	PSNR↑	***SSIM***↑	***LPIPS***↓
Beamformer	29.9909	0.8433	0.0618
Bilinear	30.807	0.8384	0.0978
INSAS	**32.6296**	**0.8654**	**0.0402**

Our *INSAS* pipeline produced introduced fewer artifacts in general compared to the beamformer method. The qualitative results in Fig.4 and Fig.2 show how the beamformer, due to the limited measurements, introduced high-frequency details that did not belong to the original scenes. On the other hand, the bilinear interpolation method has smoothened the scenes to the point where they might not be recognizable as in Fig.5 or Fig4. The average quantitative results for each angle-step scenario are detailed in Table.II and Table.I.

VI. DISCUSSION

In this work, we propose a pipeline for reconstructing SAS images from limited angular measurements using INRs. While INRs have been rising for the last few years and have been used in imaging modalities, they have not yet been largely adopted for SAS reconstruction. We believe that this work is vital for introducing and showcasing the potential of INRs in the imaging field in general and in imaging with acoustics. Despite this premise, there are a few challenges to our work that we would like to address and explore in our future works.

Firstly, while our *INSAS* pipeline surpasses the beamformer and bilinear interpolation methods in most cases, it does not take into account some of the proposed methods that could produce better results. One of these methods is deconvolving with the scene *point spread function* (PSF) [9]. Incorporating a deconvolution in the pipeline could improve the reconstruction results from sparse angular measurements.

Secondly, we experimented with one regularizer term that has not yet provided an advantage. Including another regularizer term pertaining to the physical geometry of the scene could influence the INR results with a physics-informed approach.

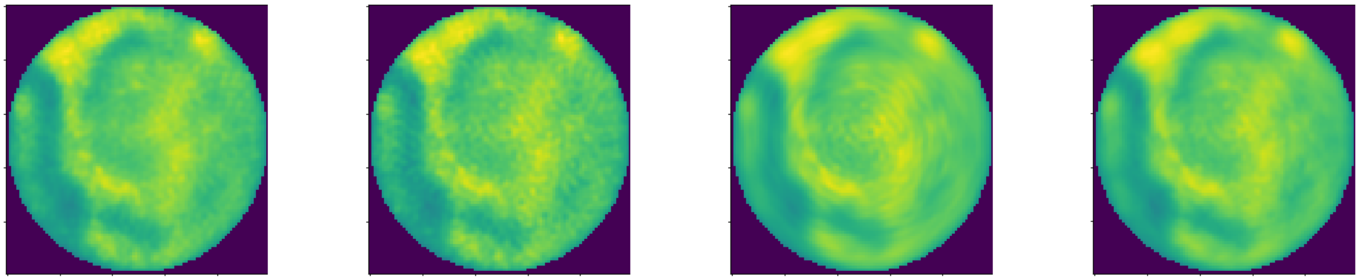

Fig. 2. Results from the eight angles step scenario. Forty-Five measurements are available as ground truth. From left to right: ground truth image, beamformer, bilinear, and *INSAS* results (ours).

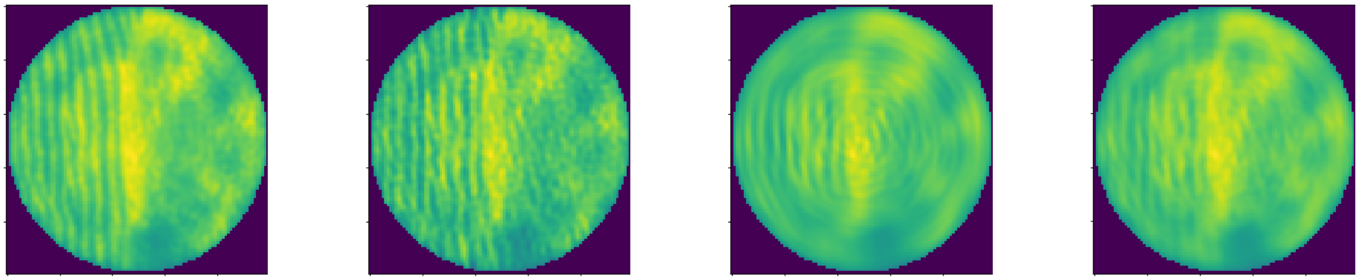

Fig. 3. Results from the twelve angles step scenario. Only thirty measurements are available as ground truth. From left to right: ground truth image, beamformer, bilinear, and *INSAS* results (ours).

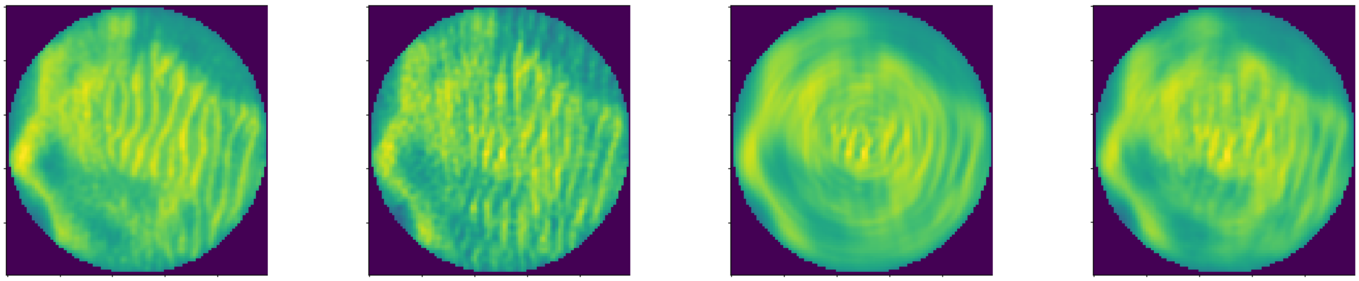

Fig. 4. Results from the sixteen angles step scenario. Only 22 measurements are available as ground truth. From left to right: ground truth image, beamformer, bilinear, and *INSAS* results (ours).

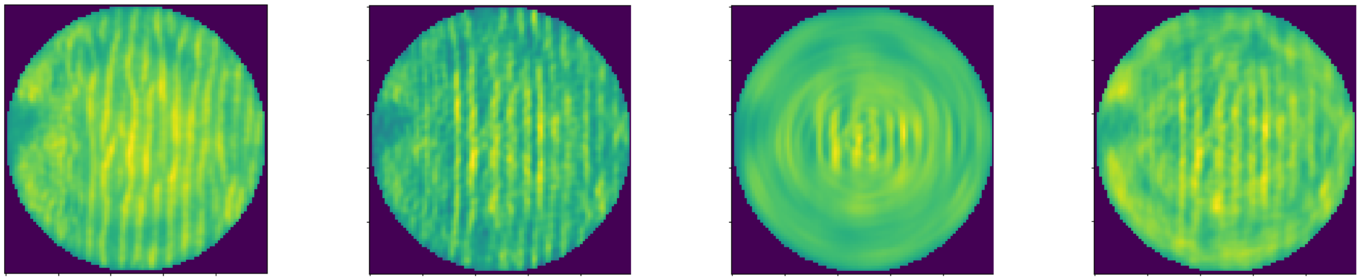

Fig. 5. Results from the twenty angles step scenario. Only 18 measurements are available as ground truth. From left to right: ground truth image, beamformer, bilinear, and *INSAS* results (ours).

VII. Future Work

For future work, we intend to experiment with real-life data captured with a SAS system. We also intend to experiment with deconvolution using multiple PSF spread across the scene. Moreover, we are also foreseeing potential gains by using a pre-trained network or a network that utilizes learned initializations [13]. We believe that these directions, along with experimentations with different regularizers, have the prospects of improving the quality of reconstructed SAS images.

References

[1] Synthetic Aperture Sonar (SAS), *Exploration Tools: Synthetic Aperture Sonar: NOAA Office of Ocean Exploration and Research.* [Online]. Available: https://oceanexplorer.noaa.gov/technology/sonar/sas.html. [Accessed: 13-Nov-2022].

[2] 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 *European Conference on Computer Vision*, pp. 405–421, Springer, 2020.

[3] Park, J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S. (2019). DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

[4] Y. Sun, J. Liu, M. Xie, B. Wohlberg and U. S. Kamilov, ”CoIL: Coordinate-Based Internal Learning for Tomographic Imaging,” in *IEEE Transactions on Computational Imaging*, vol. 7, pp. 1400-1412, 2021, doi: 10.1109/TCI.2021.3125564.

[5] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, “Deformable neural radiance fields,” arXiv preprint arXiv:2011.12948, 2020.

[6] Reed, A.W., Kim, H., Anirudh, R., Mohan, K.A., Champley, K.M., Kang, J., Jayasuriya, S. (2021). Dynamic CT Reconstruction from Limited Views with Implicit Neural Representations and Parametric Motion Fields. 2021 *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2238-2248.

[7] Strümpler, Y., Postels, J., Yang, R., Gool, L.V., Tombari, F. (2022). Implicit Neural Representations for Image Compression. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds) *ECCV* 2022.

[8] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng, “Fourier features let networks learn high frequency functions in low dimensional domains,” *NeurIPS*, 2020.

[9] Reed, A.W., Blanford, T.E., Brown, D.C., & Jayasuriya, S. (2021). Implicit Neural Representations for Deconvolving SAS Images. *OCEANS* 2021: San Diego – Porto, 1-7.

[10] Cowen, B., Park, J. D., Blanford, T. E., Goehle, G., Brown, D. C. (2021, December). AirSAS: Controlled Dataset Generation for Physics-Informed Machine Learning. In *NeurIPS Data-Centric AI Workshop*.

[11] Reed, A., Blanford, T., Brown, D. C., Jayasuriya, S. (2021, September). Implicit neural representations for deconvolving sas images. In *OCEANS 2021*: San Diego–Porto (pp. 1-7). IEEE.

[12] Zhang, R., Isola, P., Efros, A., Shechtman, E., Wang, O. (2018). The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*.

[13] Matthew Tancik, Ben Mildenhall, Terrance Wang, Divi Schmidt, Pratul P. Srinivasan, Jonathan T. Barron, Ren Ng (2021). Learned Initializations for Optimizing Coordinate-Based Neural Representations. In *CVPR*.